

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



***DISEÑO Y DESARROLLO DE UNA INTRANET DE SOPORTE
A PREVENTA***

**PROYECTO FIN DE CARRERA
INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN:
TELEMÁTICA**

Autora: María José Pérez Fernández
Tutor: Julio Villena Román

Septiembre 2009

Título: Diseño y desarrollo de una Intranet de soporte a Preventa.

Autor: María José Pérez Fernández

Tutor: Julio Villena Román

EL TRIBUNAL

Presidente:
Manuel Ureña

Secretario:
Luis de la Fuente

Vocal:
David del Valle

Realizado el acto de defensa del Proyecto Fin de Carrera el día 24 de Septiembre de 2009 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de:

Fdo: Presidente

Fdo: Secretario

Fdo: Vocal

Agradecimientos

A mi abuelo, gracias por ser así.

Resumen

En la actualidad, en el mundo empresarial está experimentando un cambio significativo en el modo de presentar y transmitir la información. El soporte en papel ha sido superado por el formato digital y la mensajería tradicional ha dejado paso a trabajar a través del e-mail. Por ello, departamentos como los de Preventa, dónde se trabaja con gran cantidad de información, la manera en que ésta se presenta es fundamental. Existen ya una gran cantidad de herramientas orientadas a empresas para tratar este aspecto. Por este motivo, se pensó en el diseño y desarrollo de un tipo de Intranet concreta para el departamento de una determinada empresa dónde se llevó a cabo este proyecto. Durante el presente proyecto fin de carrera se ha establecido una estructura básica para intercambiar información y se han desarrollado aplicaciones específicas y fácilmente sostenibles para unas necesidades concretas. Todo el desarrollo ha sido pensado para solventar problemas reales del cliente, intentando siempre que fuese práctico y robusto.

Índice general

<i>Agradecimientos</i>	<i>i</i>
<i>Resumen</i>	<i>ii</i>
1 Introducción	1
1.1 Motivación	1
1.2 Objetivos	3
1.3 Estructura del documento	4
2 Estado del arte	5
2.1 Intranet	5
2.2 Intranet Corporativa	5
2.3 Sistemas de Gestión de Contenidos.....	6
2.4 WAMP51.7.1.....	9
2.5 Apache.....	10
2.5.1 Arquitectura de Apache.....	11
2.6 MySQL.....	12
2.7 PHP.....	13
2.8 XAJAX.....	14
3 Diseño del sistema	16
3.1 Elección de herramientas para el proyecto.....	16
3.1.1 Microsoft SharePoint.....	19
3.1.2 OpenCMS.....	20
3.1.3 Plone (de Zope)	21
3.1.4 PostNuke.....	22
3.1.5 Drupal.....	23
3.1.6 Mambo.....	23
3.1.7 Joomla	25
3.1.8 Comparativa entre CMS	25
3.2 Diagramas generales del sistema	26
3.3 Perfiles de usuarios Mambo	28
3.3.1 Public Frontend	29
3.3.2 Public Backend	30
3.4 Secciones y contenidos	31
3.5 Base de datos de conocimiento y soluciones externas	32
3.6 Conclusiones del capítulo	32
4 Implementación	33
4.1 Instalación y obtención básica del sistema.....	33
4.2 Configuración de la imagen corporativa.....	41
4.3 Creación de estructuras y contenidos.....	48
4.4 Alta de usuarios.....	54

4.5	Creación de base de datos y aplicación de Soluciones Externas	56
4.6	Base de datos de conocimiento.....	61
4.7	Recarga de desplegables	74
5	<i>Implantación del sistema.....</i>	76
5.1	Uso del sistema.....	76
5.2	Estadísticas	77
6	<i>Conclusiones y trabajos futuros.....</i>	79
6.1	Conclusiones	79
6.2	Trabajos futuros.....	80
	<i>Anexos.....</i>	83
	Manual de usuario.....	83
	Para entrar	83
	Para editar contenidos	83
	Para insertar documentos	84
	Instrucciones para recuperar la Intranet.....	85
	Diagrama de clases PHP.....	87
	Soluciones Externas	87
	BBDD Conocimiento	92
	<i>Referencias.....</i>	95

Índice de figuras

Figura 1: Logotipo WAMP	9
Figura 2: Logotipo Apache.....	11
Figura 3: Funcionamiento de PHP	14
Figura 4: Tabla comparativa de CMS.	25
Figura 5: Ciclo de vida software de este proyecto.....	26
Figura 6: Diagrama de bloques de la Intranet.....	28
Figura 7: Vista de distintas posibilidades para dar de alta un usuario.....	29
Figura 8: Interfaz de acceso del administrador.....	31
Figura 9: Instalación de Wamp.....	34
Figura 10: Vista de localhost.....	35
Figura 11: Vista de Wamp instalado	35
Figura 12: Vista de sistema de carpetas en Wamp	36
Figura 13: Vista de sistemas de directorios existentes en localhost.....	37
Figura 14: Comprobación de instalación válida	38
Figura 15: Vista de WWW una vez está Mambo instalado.....	38
Figura 16: Paso 2 de la instalación de mambo	39
Figura 17: Vista de Mambo instalado con valores por defecto.	40
Figura 18: Vista del Template de Mambo que se va a usar.....	42
Figura 19: Vista de la edición del FrontPage.	43
Figura 20: Vista de Mambo con algunos cambios realizados.	45
Figura 21: Vista de Mambo con la CSS modificada.	45
Figura 22: Vista de Mambo con el Template definitivo.....	48
Figura 23: Vista de una sección sin categorías.....	49
Figura 24: Vista de un contenido.....	51
Figura 25: Vista de proceso de creación del Template.....	52
Figura 26: Vista de modificaciones del Template.	53
Figura 27: Vista del Template con estructura creada.	54
Figura 28: Vista de usuarios.	55
Figura 29: Vista del Template por un usuario tipo editor.....	55
Figura 30: Esquema de tablas de Soluciones Externas con sus claves.....	58
Figura 31: Vista del interfaz de la base de datos, tabla Soluciones.....	58
Figura 32: Lista de soluciones tipo PABX.	59
Figura 33: Vista en el caso de que no existan soluciones PABX.....	60
Figura 34: Vista de un integrador concreto.	60
Figura 35: Imagen de cómo incrustar el código PHP.....	65
Figura 36: Código PHP incrustado en el Template.	66
Figura 37: Contenido sin que este correctamente hecho el mosaddphp.....	66
Figura 38: Subida correcta de ficheros e inserción en la base de datos.....	72
Figura 39: Comprobación de fichero copiado en la carpeta de destino.....	72
Figura 40: Comprobación de posibilidad de descarga de un fichero subido por los propios usuarios.....	74
Figura 41: Vista de formulario para insertar proyectos.....	75
Figura 42: Imagen de estadísticas que se pueden obtener.	78
Figura 43: Clases PHP involucradas en la sección "Soluciones Externas".....	88
Figura 44: Vista de enlace que lleva al archivo PHP.	89
Figura 45: Listado de soluciones de la categoría "CABLEADOS E INFRAESTRUCTURAS"	90

Figura 46: Datos del Integrador seleccionado.	91
Figura 47: Ficheros PHP involucrados en la categoría "Soluciones Externas".....	92
Figura 48: Ficheros PHP involucrados en la categoría "Proyectos".	93
Figura 49: Ficheros PHP involucrados en la categoría "EVS".....	93

1 Introducción

1.1 Motivación

La realización de este proyecto se sitúa dentro del departamento de Ingeniería Preventa de una empresa multinacional, France Telecom España S.A. Este departamento se encarga de diseñar soluciones tecnológicas a medida para otras empresas. Los clientes que demandan una determinada estructura de telecomunicaciones presentan un pliego de condiciones de qué es lo que necesitan y varias empresas (entre la cuales se sitúa donde se ha hecho el proyecto) se disponen a realizar el mejor diseño posible en función de las necesidades del cliente. Una vez que cada empresa tiene su proyecto lo presenta al cliente y es éste quién decide que empresa será la encargada de llevar a cabo el proyecto.

Desde el momento en que la empresa recibe el pliego de condiciones del cliente, los ingenieros se ponen a realizar la propuesta que finalmente se presentará. Primero pasarán a otro departamento un estudio de viabilidad (EV) para que otros ingenieros confirmen si en la zona geográfica donde se tiene que implementar la solución se dispone de ciertos recursos (fibra, radio...). Una vez conocidos los recursos con los que se cuenta, se lleva a cabo la propuesta. En dicha propuesta pueden intervenir diferentes tecnologías y de última generación. Para ello disponen de soluciones tecnológicas pertenecientes a la empresa y otras soluciones externas que ofrecen distintos proveedores.

Todo este proceso genera una gran cantidad de información: documentos, bases de datos, mapas, estudios de viabilidad (EV), presentaciones, presupuestos, mejoras...

Previamente a este proyecto, esta información estaba repartida por distintas unidades de red corporativas, pero sin seguir un criterio u orden concreto. A estas unidades de red tenían acceso todas las personas dentro del departamento. El procedimiento que seguía hasta el momento cada vez que alguien disponía de un documento con cierto valor para el futuro era altamente ineficaz. Ese documento se colocaba dentro de una carpeta que había para el departamento, pero que podía contener miles de documentos sin orden alguno. Como esto podía ser tedioso, otra posibilidad era enviar un correo electrónico adjuntando dicho documento a quien le pudiese interesar. Esta situación implicaba a que cuando se fuese a buscar algún documento de ayuda, no se supiese dónde ir y se perdía mucho tiempo

buscando por las unidades. Además con este sistema no se sabía cuándo alguien había colocado algún documento nuevo en dichas unidades.

Sin embargo esta información puede facilitar y agilizar de gran manera el trabajo si se puede acceder a ella de manera fácil y ordenada. Por ejemplo, antes de empezar con un nuevo proyecto, ver qué proyectos ha realizado ya la empresa para ese cliente, con respecto a esa tecnología o con ese proveedor. También así se puede ver al instante la calidad de las soluciones implantadas, qué proyectos se ganaron o se perdieron, qué estudios de viabilidad fueron rechazados y por qué...

Por lo tanto el problema a solucionar era este. Era la necesidad de crear un contexto que facilitase tanto el acceso como el intercambio de conocimientos. De lo que se trataba era que cualquier ingeniero tuviese un lugar ‘amigable’ donde encontrar todo lo necesario para su trabajo diario. Un lugar donde encontrar las últimas noticias que a ellos les afectasen, como relaciones con otros departamentos, nuevas medidas a aplicar, documentación técnica... Esta estructura debía contar con un apartado especial donde cualquier ingeniero pudiese tanto insertar como buscar documentos que pudiesen ser reutilizables en un futuro. Este apartado debería comprender tres tipos de documentos distintos:

- Las soluciones externas existentes, con información relativa al escenario de aplicación, productos relacionados...Aquí también se incluiría información entre otras cosas de cada proveedor o proveedores que las realizasen con su información específica correspondiente.
- Propuestas de proyectos realizados con anterioridad (haya sido ganado o no), con referencias al documento, a los ingenieros que trabajaron en el proyecto, al tipo de tecnologías utilizadas, fecha de presentación del proyecto...
- Estudios de viabilidad para poder ver rápidamente en qué zonas hay determinados servicios. Esto en ocasiones puede resultar más rápido que hacer una petición a otro departamento. En este apartado aparecerían datos correspondientes a los ingenieros

1.2 Objetivos

Por tanto, el objetivo del proyecto es el diseño y desarrollo de una Intranet corporativa para el departamento como repositorio de información que solucionase todos los problemas anteriormente comentados. Con la Intranet se pretendía tener ordenada, accesible y bien presentada toda esa información. Esta Intranet debería cumplir distintas necesidades, y ser lo suficientemente flexible para hacer las modificaciones futuras que los usuarios consideren de utilidad en su trabajo. Las principales necesidades eran un lugar donde al conectarse estuvieran informados en cada momento de las últimas noticias que pudiesen afectar, poder buscar e insertar información de tareas pasadas y que toda esta información sólo la pudiesen ver los trabajadores a los que les afecta. También es muy interesante que sea interactiva con los propios usuarios de dicha herramienta ya que probablemente ellos tengan una idea más clara de lo que necesitan y de qué tareas pasadas fueron más importantes. Sin llegar a crear una Wiki completa, sí por lo menos que pudiesen dejar reflejada su opinión sobre ciertos temas.

Entonces se decidió crear una Intranet que iba a estar orientada a crear, almacenar, recuperar, intercambiar y usar los conocimientos de los que la empresa dispone. A través de ella se iba a comunicar y almacenar información de forma diferente a lo que se había hecho hasta el momento, permitiendo visualizar y comunicar conocimientos a otras personas.

Por una parte se pretendía tener una parte básica de la Intranet con las funcionalidades mínimas para cualquier Intranet que se pudiese montar en no demasiado tiempo. Además era importante que esta parte básica pudiese ser compatible con otras tecnologías con las que desarrollar después, dedicándole más tiempo a una o varias aplicaciones a medida para el departamento. Al desarrollar estas aplicaciones se aprovechan las ventajas que ofrece esta tecnología Web.

Era muy importante que tuviese un mantenimiento sencillo, alguna manera de hacer copias de seguridad de todo lo que allí había y que los usuarios trabajasen fácilmente con ella. Hacer que los usuarios fuesen utilizándola poco a poco hasta llegar a considerarla una herramienta relacionada con gran parte de las actividades que se llevan a cabo dentro de la organización. También era importante que no generase demasiados gastos adicionales, tuviese una interfaz corporativa y lo más cómodo posible.

Por motivos de coste de propiedad, otro de los objetivos era encontrar una solución basada en software libre.

1.3 Estructura del documento

El esquema de esta memoria es el siguiente:

- En el capítulo 1, la presente introducción, donde se sitúa el contexto concreto del proyecto. Aquí se explica, por que motivos era necesaria la realización del proyecto y cuáles fueron los objetivos fundamentales marcados, así como algunas directrices para la realización del mismo.
- En el capítulo 2, se presenta el estado del arte de todas las tecnologías con las que se ha trabajado a la hora de realizar el proyecto. Se explican las características técnicas de cada una de ellas, así como sus diferentes tipos.
- En el capítulo 3, se explica el diseño del sistema, lo que conlleva a explicar la investigación previa para decidir las tecnologías a utilizar. Es decir, se explican las tecnologías de las que se disponía como posibles soluciones al proyecto, y el porqué se descartaron unas y se escogieron otras. En este apartado hay una comparativa entre diferentes CMS.
- En el capítulo 4, es donde se detalla la implementación de todo el proyecto. Es decir, los pasos que se deben de seguir y que se debe de hacer para obtener la Intranet tal y cómo ahora está funcionando.
- En el capítulo 5, se habla de la implantación del sistema. Aquí se explica cómo es el uso dado por los usuarios a esta herramienta y su frecuencia de uso. Además, hay un apartado que hace referencia a unas estadísticas en las que se podía comprobar cualquier dato estadístico de uso de esta herramienta (mes en el que más se usa, hora punta de uso...).
- En el capítulo 6, se desarrollan tanto conclusiones como futuros trabajos. Por una parte, se explican los objetivos cumplidos del proyecto y beneficios aportados y por otra, se comentan posibles ampliaciones o mejoras teniendo como base la Intranet.
- Por último, se incluyen anexos y referencias. En los anexos se incluye un manual de usuario, dónde se le explica al usuario las diferentes tareas que puede realizar: acceder a la Intranet, editar contenidos e insertar documentos. En las referencias, aparecen todas las fuentes consultadas para la realización del presente proyecto.

2 Estado del arte

2.1 Intranet

Una Intranet es una red de ordenadores privada basada en los estándares de Internet [9]. Su objetivo es favorecer la comunicación interna y la transmisión de información dentro de la propia organización ayudando con esto a disminuir costes.

Está basada en el protocolo TCP/IP y habitualmente dispone de un sistema firewall para no permitir el acceso a la red desde el exterior. Se trata de una red aislada, no conectada abiertamente a Internet. Su principal diferencia con Internet, es que Internet es una red abierta, de acceso público mientras que la Intranet se restringe únicamente a una organización. Una Intranet puede extenderse a través de Internet, lo cual se hace usando una red privada virtual (VPN).

Una Intranet es una herramienta realmente útil para combatir la pérdida de tiempo, esfuerzo e información entre los trabajadores de una misma organización. Una de sus principales características tal vez sea que es un sistema fácil de manejar que permite generar información de mayor calidad y de forma más rápida. Esta información se transmite de forma más rápida y más flexible que el papel tradicional. Esto implica también una reducción en el coste del papel, ya que mucho del material impreso se vuelve obsoleto.

2.2 Intranet Corporativa

Los elementos indispensables [7] para poner en funcionamiento una Intranet Corporativa no son muy costosos:

1. Un ordenador que será usado como servidor Web. Al depender de forma directa la Intranet de este equipo, conviene que este se parezca en todo lo posible a una maquina bastión [25], es decir que en la medida que sea posible el PC funcione a buen rendimiento la mayor cantidad de tiempo posible. Tiene especial importancia en este ordenador elementos como antivirus, antiespías y cualquier medida de seguridad.

Es conveniente que la copias de seguridad que se hagan, se encuentren en este equipo en local y en otro externo, por seguridad.

2. Puestos de trabajo existentes que deberán contar con programas cliente “ligeros” (navegadores). La Intranet se debería poder visualizar en la mayor cantidad de navegadores posibles.
3. Instalación de una red de comunicaciones LAN/WAN. En este caso en la organización ya existía una red así antes de montar la Intranet.
4. Conjunto estandarizado de protocolos: IP, HTTP, SMTP...
5. Software apropiado.
6. Datos y conocimientos relevantes que se quiera compartir.
7. Personal que diseñe y actualice las páginas Web.

2.3 Sistemas de Gestión de Contenidos

Un sistema de gestión de contenidos [18] (CMS, Content Manager System en inglés) es un programa que permite crear una estructura de soporte para la creación, administración y compartición de distintos tipos de contenidos por parte de los usuarios y el webmaster de un sitio Web. Se trata de un interfaz que maneja una o varias bases de datos donde se encuentran la información de la aplicación. Un CMS tiene dos funciones principales: facilitar la creación y la presentación de esos contenidos. Esto lo hace administrando de forma independiente ambas características: el diseño y el contenido.

Provee las herramientas necesarias para gestionar el ciclo de vida de los contenidos: creación, gestión, presentación, mantenimiento y actualización. Los wikis y los sistemas groupware también son considerados CMS.

Un buen CMS permite, por ejemplo, a los empleados sin demasiados conocimientos sobre este tema publicar páginas dinámicas, formularios, formaciones y todo tipo de contenidos desde programas de edición como Microsoft Word.

La generación, publicación y archivo de un número enorme de páginas en los portales y sitios Web plantea retos que los CMS intentan resolver:

- Facilitar la generación y edición de contenidos para la Web por parte de personal sin formación específica en programación.

- Asegurar una apariencia homogénea de todos los contenidos y su presentación de acuerdo con un diseño corporativo y con una línea editorial predefinida.
- Mantener una coherencia en la estructura de los sitios Web, permitiendo la incorporación de nuevos contenidos en las secciones adecuadas tras la aprobación y el control previo de las personas indicadas.
- Mantener una navegación consistente que permita a los usuarios llegar a cada uno de los contenidos que están publicados en cada momento.
- Evitar la existencia de contenido duplicado (distintas direcciones URL que muestran un mismo contenido), contenido huérfano (archivos que quedan en el servidor de forma innecesaria ya que son páginas a las que ya no apunta ningún enlace o archivos de imagen o multimedia que se mostraban en páginas suprimidas) o bien enlaces rotos, que apuntan a páginas que no existen en el servidor.

Los gestores de contenidos o CMS son herramientas de software que permiten descentralizar las labores de mantenimiento del contenido de un portal, de forma que personal no técnico de los distintos departamentos de una empresa puede añadir, editar y gestionar su propio contenido en una Web corporativa.

En definitiva, entre los usos del Sistema de Gestión de contenidos se podían mencionar:

1. Publicar contenidos de forma sencilla.
2. Creación de la estructura de la Web.
3. Control de usuarios.
4. Interacción con los usuarios.
5. Seguridad.
6. Adaptación a los estándares y a las nuevas tecnologías.

Existen versiones gratuitas y versiones propietarias.

Las versiones gratuitas tienen como características:

- Coste de propiedad muy reducido.
- Acceso libre al código de la aplicación.
- Comunidad de usuarios.

- Intercambio de información.
- A veces, documentación escasa.

En cuanto a las versiones propietarias:

- Pago por licencias de uso.
- Acceso restringido al código de la aplicación.
- Empresa.
- Soporte profesional.
- Documentación habitualmente correcta.

Cabe mencionar diferentes tipos de CMS:

- Genéricos: Disponen de la plataforma necesaria para desarrollar e implementar aplicaciones que dan soluciones a necesidades concretas. Ejemplos: Zope, MODx, OpenCMS, TYPO3, Apache Lenya, Joomla, Drupal...
- Foros: lugar que permite conversaciones en línea donde los usuarios pueden reunirse y discutir sobre algunos temas en concreto en los que están interesados.
- Blogs: Publicación online con historias publicadas con una periodicidad muy alta que son presentadas en orden cronológico inverso, es decir, lo último que se ha publicado es lo primero que aparece en la pantalla.
- Wikis: Páginas Web con enlaces, imágenes y cualquier tipo de contenido que puede ser visitada y editada por cualquier persona. Ejemplo: Wikipedia
- eCommerce: Distribución, compra, venta y suministro de información complementaria para productos o servicios a través de redes informáticas como Internet u otras.
- Portal: Es un lugar central en que se puede poner todo tipo de información para un público muy diverso.
- e-Learning: Es un tipo de educación virtual a distancia que se realiza a través de Internet.
- Publicaciones digitales: Publicación electrónica y sin papel realizada a través de las nuevas tecnologías.

En el apartado 3.1 se hace una descripción detallada de varios CMS, evidentemente entre ellos Mambo.

2.4 WAMP51.7.1

Por su interés para el desarrollo de este proyecto, se va a describir WAMP y las tecnologías en las que se basa [10].

WAMP es el acrónimo de Windows, Apache, MySQL y uno o más lenguajes de Script que pueden ser PHP, Perl y/o Python. Es un paquete de software como LAMP en Linux, pero en este caso para Windows.



Figura 1: Logotipo WAMP

Se trata de un software que permite instalar de una sola vez y de forma automática, todo lo necesario para crear un sitio Web en un servidor local. Es de código abierto (open source), libre (licenciado bajo GPL) y gratuito. Tiene una aplicación para manejar los servicios que se instala como un “tray icon”.

Con él se cubren todos los requisitos mínimos para un sitio Web:

1. Servidor Web→Apache
2. Entorno de programación→PHP5
3. Base de datos→MySQL
4. phpMyAdmin→código en PHP para administrar una base de datos MySQL utilizando para ello un interfaz Web.

En teoría, ya con esto es suficiente para crear las páginas PHP directamente. Sin embargo, en este proyecto se deseaba encontrar algún tipo de software complementario a este, que proporcionase ciertas funcionalidades ya implementadas que pudieran ser de utilidad a la hora del desarrollo de una Intranet.

Fue entonces cuando se pensó en un CMS (o Sistema de Administración de Contenidos), ya que consiste en una interfaz que puede controlar varias bases de datos donde se encuentran los datos de una página Web. En la actualidad existen gran cantidad de CMS

como Joomla, pero se pretendía encontrar uno del cual existiese la mayor cantidad de información disponible para trabajar con el.

Como Joomla es una versión más reciente de Mambo, se pensó que sería mejor el uso de Mambo ya que se dispondría de mayor información sobre el.

A continuación se pasa a describir en mayor detalle los distintos componentes en que se basa WAMP: Apache, PHP y MySQL.

2.5 Apache

Es un servidor Web [27], de tecnología Open Source, sólido y para uso comercial desarrollado por la Apache Software Foundation. Es uno de los más utilizados en Internet. Sólo le hace competencia el servidor de Microsoft IIS.

Algunas de sus características son:

1. Servidor Web rápido, flexible y eficiente.
2. Está continuamente actualizado y adaptado a los nuevos protocolos (http 1.1, WebDav...).
3. Implementa los últimos protocolos aunque se base en http 1.1.
4. Modular: Fácil adaptación a diferentes protocolos y necesidades, con diferentes módulos de apoyo y con la API de programación de módulos. Actualmente existen muchos módulos Apache que son adaptables a este y que están ahí para ser instaladas cuando sea necesario. Otro aspecto destacable es que cualquiera que posea una experiencia mínima en programación C o Perl puede escribir un modulo para realizar una función determinada.
5. Promueve la comunicación con el usuario con el fin de obtener nuevas ideas, informes de fallos y parches para solucionar los mismos.
6. Multiplataforma.
7. Extensible: debido a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, lenguaje de programación del lado del servidor.
8. Se ejecuta en una multitud de sistemas operativos.

9. Apache trabaja con Perl, PHP y otros lenguajes de Script. También trabaja con Java y páginas JSP. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
10. Tiene una alta configurabilidad en la creación y la gestión de logs. Apache permite la creación de ficheros de Log a medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

Si se compara Apache con otros servidores de software libre ha quedado demostrado que Apache es el más rápido. Aunque de la comparación de Apache con servidores comerciales pueda resultar que Apache no es el más rápido, no merece la pena gastarse miles de dólares en un servidor por una diferencia no tan grande en rapidez.



Figura 2: Logotipo Apache

2.5.1 Arquitectura de Apache.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos de Apache se pueden clasificar en tres categorías:

- Módulos Base: Módulo con las funciones básicas del Apache. Ejemplo: core.
- Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones. Ejemplo: mpm_common, beos, worker...
- Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor. Ejemplo: mod_access, mod_actions, mod_alias...

2.6 MySQL

MySQL es un sistema gestor de bases de datos relacionales en SQL [8], esto significa que permite la gestión de los datos de una BBDD relacional usando un lenguaje de consulta estructurado. Y por tanto, a partir de una consulta, MySQL llevará a cabo una determinada acción sobre una base de datos. Es un excelente gestor de bases de datos que la sitúan después de Oracle como mejor solución a nivel técnico por las características explicadas a continuación.

Sus principales características son:

- **Velocidad:** Por estudios recientes se puede afirmar que MySQL es un sistema con una velocidad superior a sus rivales, incluido Oracle a la hora de ejecutar consultas SQL.
- **Portabilidad:** Puede ejecutarse en la inmensa mayoría de los sistemas operativos, por lo que junto a un lenguaje de programación del lado del servidor de alta portabilidad como Java, PHP, Perl...etc., permite el desarrollo de aplicaciones Web fáciles de migrar y el acceso y copia de los datos desde cualquier Sistema Operativo.
- **Seguridad:** Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el Host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptada cuando se conecta con un servidor.
- **Escalabilidad y límites:** Soporte a grandes bases de datos. Se usa con bases de datos que tienen 50 millones de registros y con 60.000 tablas. Se permiten hasta 64 índices por tabla.
- **Funcionalidad:** Tal vez este sea el punto en que más desventaja tenga con sus rivales. Sin embargo dispone de muchas funciones vitales como duplicación, volcado online...etc.
- **Conectividad:** Los clientes pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma.
- **Localización:** El servidor puede proporcionar mensajes de error a los clientes en muchos idiomas.

2.7 PHP

PHP es el acrónimo de Hypertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma [19], [20], [21], rápido con una gran librería de funciones y mucha documentación.

Es un lenguaje de programación interpretado utilizado para la creación de páginas Web dinámicas, ampliamente usado y que está diseñado especialmente para desarrollos Web y puede ser integrado dentro de código HTML. Aunque su principal función es la creación de páginas Web también puede ser utilizado para crear aplicaciones con una interfaz gráfica para el usuario y también puede ser usado desde la línea de comandos. Permite la conexión a distintos tipos de bases de datos como: MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. Puede ser ejecutado en la mayoría de los sistemas operativos como UNIX (Linux, MAC OS) y Windows. Puede interactuar con los servidores Web más populares ya que existe en versión CGI, módulo para Apache e ISAPI. Algunas características son:

- No necesita definición de tipos de variables.
- Tiene manejo de excepciones.
- Todo el mundo puede acceder a él puesto que es libre
- Dispone de una gran documentación
- Multiplataforma.
- Es un lenguaje más cercano a JavaScript o a C más que a lenguajes de marcas como podrían ser XML, HTML o WML. Pero su principal diferencia con JavaScript o C que se ejecutan en el navegador, PHP se ejecuta en el servidor. Esto permite acceder a cualquiera de los recursos que tenga el servidor como por ejemplo la base de datos. Por lo tanto el programa es ejecutado en el servidor pero el resultado de esta ejecución es enviada al navegador. Este resultado normalmente es una página HTML pero también podría ser una página WML.
- Al ser PHP un lenguaje que se ejecuta en el servidor y que es independiente del navegador no es necesario que el navegador del cliente lo soporte. Aunque si es necesario que para que las páginas PHP funcionen el navegador del servidor si lo soporte.

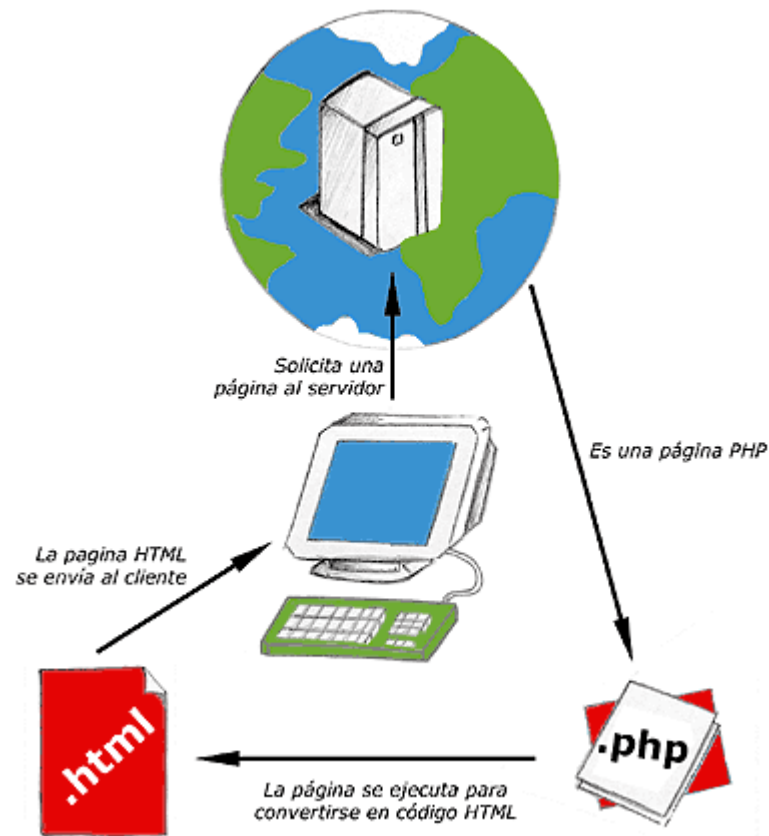


Figura 3: Funcionamiento de PHP

2.8 XAJAX

XAJAX o también conocido como AJAX para PHP es sencillamente una librería PHP que contiene la clase denominada XAJAX. Con esta librería se consigue mayor sencillez en el manejo de AJAX con PHP, utilizando para ello sentencias JavaScript [1].

XAJAX se usará en formularios con varios desplegaables de tal forma que cada vez que se seleccione un desplegable solo se recargue esa parte del formulario manteniendo la información que el usuario ha completado en otras partes del formulario.

Algunas características de XAJAX son las siguientes:

- Compatible con gran cantidad de navegadores de Internet: Firefox, Mozilla, Internet Explorer, Opera y Safari

- Puede ser usado para actualizar estilos, clases CSS, botones de selección, casillas de verificación y botones de radio o cualquier otro atributo de un elemento.
- Cada función registrada para ser accesible a través de XAJAX puede tener distintos tipos de petición.

En una aplicación AJAX, el servidor crea una página que es enviada al cliente. Éste interactúa con la página de alguna forma como rellenando formularios, haciendo clicks en determinados objetos... Y estos hechos hacen que se disparen ciertos eventos como onclick, onchange que llaman a funciones JavaScript. Estas funciones pueden o no interactuar con el servidor y recibiendo información de éste mostrándola al usuario cambiando el contenido de la misma página (una parte del contenido, no pidiendo una página nueva).

XAJAX permite escribir funciones en PHP que pueden ser accedidas cuando ya la página ha sido enviada al navegador, cuando el usuario ha provocado que se dispare un evento o la función PHP ha sido llamada desde un JavaScript. Estas funciones PHP modifican la apariencia de la página Web y como se hace desde PHP existen posibilidades tales como acceder a bases de datos, a sistemas de archivos y todo esto sin necesidad de recargar toda la página.

Este tipo de páginas son muy útiles y son uno de los pilares de la Web 2.0.

3 Diseño del sistema

3.1 Elección de herramientas para el proyecto

A la hora de diseñar el sistema hay que tomar decisiones sobre varios elementos.

Uno de ellos será el sistema operativo en el que se trabajará. Como el entorno de la empresa donde se realizará el proyecto está completamente basado en Windows será lógico elegir éste como sistema operativo a utilizar.

Como se pretende desarrollar funcionalidad añadida a la Intranet, habrá que decidir en que lenguaje de programación se va a desarrollar. Además, esta elección del lenguaje va a condicionar la elección del software de gestión de contenidos que más se adapte según las necesidades. Buscando información relativa a lenguajes de programación aptos para crear páginas Web se encuentran los siguientes:

- Perl → es un lenguaje de programación interpretado que toma que toma varias características del lenguaje C [28]. Su principal rasgo es que no tiene ninguna limitación que sí tienen otros lenguajes de script. Es decir, posee una gran libertad sintáctica. Éste no es considerado un lenguaje interactivo, por lo que es el más conservador de los tres. Tradicionalmente ha sido utilizado para la administración de sistemas, sobre todo para el tratamiento de la información de texto.
- PHP→ lenguaje de programación interpretado, diseñado desde un principio para la creación de páginas Web dinámicas [20]. Este lenguaje se integra con la lógica de las páginas. Los usos de PHP son casi exclusivamente la creación de páginas Web y además es más fácil que con los otros dos.
- Python→ es un lenguaje interpretado que ahorra tiempo considerable en el desarrollo del programa ya que no necesita compilar ni enlazar [23]. Tiene una consola desde la cual se pueden escribir los comandos, a diferencia de Perl. De estos tres Python está totalmente orientado a objetos, los otros dos se pueden utilizar de forma orientada a objetos o no. Contiene elementos de programación funcional. Su característica principal está en programar fácil y

comprensiblemente. Tiene reglas estrictas, al contrario que Perl. Permite dividir el programa en módulos reutilizables por otros programas Python. Algunos de sus usos actuales son Web Services, compilación para sistemas empujados, integración con Java...

Tanto Perl como Python están más enfocados a la administración de sistemas, mientras que con PHP se dan más facilidades a la hora de crear páginas Web.

Por estos motivos se decidió que el lenguaje de programación utilizado para la parte que se programará sea PHP.

Otro de los aspectos a utilizar será el soporte a utilizar. Se buscará el software más completo posible, con fácil mantenimiento e instalación, pero que a su vez permita crear proyectos nuevos a partir de él de una cierta dificultad. Para esto se contemplarán opciones de software libre para disminuir los costes lo máximo posible.

Primero habrá que resolver qué software usar como base sobre la que construir la Intranet. Es decir, los elementos básicos que pueden dar soporte a distintos tipos de Intranet: un servidor Web, un entorno de programación y una base de datos. Estos elementos se podrían obtener fácilmente cada uno por separado y proceder a la instalación e interacción de todos ellos. Pero se buscará un paquete que a ser posible incluya todas estas necesidades juntas y que por supuesto tenga utilidad en Windows ya que es el sistema operativo a utilizar. Se buscará que primase la sencillez y la rapidez.

De la búsqueda de paquetes que incluyesen lo anteriormente comentado se encontraron gran cantidad de ellos:

- Uniform Server 3.5 → Es la manera más pequeña para ejecutar un servidor WAMP (un solo archivo de 5,79Mb). A pesar de los pocos bytes que ocupa, funciona y hasta tiene un completo panel de control [29]. Este miniservidor puede ser de mucha utilidad para hacer pruebas en PCs que no tienen un servidor instalado, o simplemente para probar cosas sin “contaminar” el Apache de cabecera. A pesar de incluir todo lo necesario, el programa no necesita instalación por lo que ciertos parámetros que solamente se pueden introducir en el momento de la instalación no queda muy claro cuándo se introducirán. Por este motivo fue descartado.
- XAMPP → Como el anterior, contiene todo lo necesario para crear una plataforma de contenidos [1]. Pero este proyecto fue realizado pensando en un

principio en un sistema operativo Linux, aunque en la actualidad tiene versiones para distintas distribuciones. Además de los elementos necesarios para crear la base de la Intranet, posee además PEAR, Perl, GD, expat, Ming, expat... y otra serie de aplicaciones que en este caso no serían necesarias, por lo tanto se descartó por tener bastantes más funcionalidades de las estrictamente necesarias.

- EasyPHP→ Paquete para Windows [6] que contiene las funcionalidades mínimas comentadas anteriormente y que tiene la ventaja de ser un sitio español.
- Wamp Server 1.7.1 → Este paquete contiene los elementos básicos que se necesitan [10]. Este paquete instalará automáticamente Apache, PHP5, MySQL, PHPMyAdmin y SQLiteManager en el ordenador. La principal ventaja de este paquete frente al resto de paquetes de este tipo es ofrecer la versión de PHP5 en el instalador y sus constantes actualizaciones. Además los servicios de Apache y MySQL se instalarán en Windows bajo los nombres de wampapache y wampmysqld para no interferir con otras instalaciones previas de estos servidores.

Se escogió WAMP 1.7.1 por todos estos motivos, aparte de ofrecer ventajas tales como la consola unificada que permite ver en un momento el estado de todos estos componentes.

A partir del momento en que se decide la base sobre la que construir la Intranet, hay que decidir que tipo de Intranet se requiere que sea. Para decidir entre CMS o Wiki como estructura de la Intranet, hay que analizar que características se quiere que primen sobre todas las demás. En este caso tendrá que primar la buena organización y ordenación de los contenidos, por encima de la interacción con el usuario. Por estos motivos se decidirá un CMS en lugar de una Wiki, aunque sin abandonar del todo la idea de la interacción con el usuario. Existirá una interacción con el usuario, pero no será ésta su principal función.

Decidido que se va a trabajar para la creación de la Intranet con un CMS habrá que decidir con cuál quedarse de todos los que ofrece el mercado. A continuación se detallan una lista de CMS usados en Windows.

3.1.1 *Microsoft SharePoint*

Gestor de contenidos propiedad de Microsoft [12]. Este CMS comprende tanto la plataforma como otros servicios. La plataforma es Windows SharePoint Services (WSS), que viene incluida con Windows Server y está disponible como descarga gratuita para las personas con licencia de Windows Server. Esta plataforma proporciona todos los elementos que son necesarios en todas las tecnologías de Microsoft SharePoint. WSS es una estructura de objetos para crear y desarrollar páginas Web, basada en la gestión y publicación de documentos y contenidos. WSS está pensada únicamente para servidores de Microsoft y no se puede implementar en otro sistema operativo. WSS también proporciona soporte para blogs y wikis.

SharePoint ofrece contenidos a través de sitios Web de Internet Information Services (IIS). IIS es un servicio que convierte un ordenador en un servidor de Internet o Intranet. Por lo tanto los equipos que tengan instalado este servicio podrán publicar páginas Web tanto local como remotamente. IIS utiliza la tecnología SQL Server para almacenar el contenido.

SharePoint incluye aplicaciones Web tales como, espacios comunes de trabajo, paneles de control, herramientas de navegación, descripciones (incluyen e-mails), calendario compartido, listas de contactos y foros de discusión.

Los sitios Web pueden ser configurados para mostrar el contenido por separado, es decir para Intranet, Extranet y sitios de Internet. SharePoint utiliza un modelo de permisos similares a LDAP. Una página de SharePoint se construye mediante la combinación de otros elementos Web en una única página, a la cual se accede mediante el navegador. Cualquier editor Web que soporte ASP.NET es válido, pero el recomendado por el fabricante es Microsoft Office SharePoint Designer.

Las páginas WSS son aplicaciones ASP.NET y la Web SharePoint utiliza la infraestructura de ASP.NET y usando los APIS de ASP.NET algunas funcionalidades de la Web pueden ser modificadas para ampliar la funcionalidad de WSS.

Como hecho importante se considera el lanzamiento de una versión beta de WSS 3 en 2006. Esta versión daba mayor soporte a soluciones Web 2.0 como blogs, wikis y RSS.

Microsoft Office SharePoint es una aplicación que proporciona características de administración de contenido. Mediante el uso de plantillas y otras características de Office SharePoint Server, pueden crearse sitios que admitan la publicación de contenido específico, administración de contenido o suplir necesidades de inteligencia empresarial. Por ejemplo se podrían crear portales de organización con repositorio de contenido o áreas de reuniones, con

datos tales como fecha y lugar, asistentes, acta de la reunión...Esto permite colaborar y compartir información con gente que este tanto dentro como fuera de la organización. Se puede usar Office SharePoint Server para:

- Colaboración con el resto de personas de la organización.
- Crear espacios personales, para administrar y compartir información con los compañeros.
- Buscar personas, conocimientos y datos dentro de las aplicaciones empresariales.
- Publicar fácilmente, informes, listas e indicadores clave de rendimiento. Estos se podrán vincular a aplicaciones como SAP, Siebel y Microsoft SQL Server.

Tal vez debería de quedar más clara la diferencia entre Office SharePoint y Windows SharePoint. Office SharePoint es un producto que usa las tecnologías de SharePoint. Cualquier característica disponible en SharePoint también lo está en Office SharePoint. Office SharePoint está basado en Windows SharePoint. Sin embargo, Office SharePoint ofrece funciones mejoradas o adicionales que no están disponibles en SharePoint.

3.1.2 *OpenCMS*

OpenCMS es un Gestor de Contenidos Open Source [16] que ayuda a crear y mantener sitios Web completos de forma sencilla y sin necesidad de conocimientos HTML. Es distribuido por la empresa Alkacon Software bajo licencia LGPL. Es uno de los CMS más antiguos y está basado en Java, Tomcat, XML y MySQL. OpenCMS integra un editor WYSIWYG que permite escribir un documento viendo directamente cuál va a ser el resultado final. Algunas de sus características son:

- Soporte multi-idioma.
- Sistemas de gestión de usuarios y permisos integrados.
- Publicación de contenidos.
- Mecanismos de plantillas JSP y XML.
- Sistema de ayuda on-line.
- Publicación dinámica y estática de contenidos.
- Gestión de workflow.
- Entorno de trabajo basado en navegador.
- Versiones del contenido.

- Sistema de caché.
- Exportación de contenido dinámico y estático.

En OpenCMS existen tres formas de presentar los contenidos a los visitantes Web: estático, dinámico y mixto. El entorno de trabajo de OpenCMS es dinámico. Los archivos y carpetas son creados, editados, bloqueados y desbloqueados en el Workplace (Intranet). OpenCMS permite que cada recurso pueda ser configurado independientemente.

Tiene dos tipos de proyectos:

- Offline: es el proyecto de trabajo de OpenCMS, es decir, aquel en el que se encuentra siempre que se está trabajando sobre los recursos de OpenCMS. Normalmente es el modo en el que se trabaja. De esta manera se puede trabajar tranquilamente sobre los contenidos, sin que en el exterior se estén observando los cambios que se están realizando. Es una manera de asegurarse de que la información no se hace pública hasta que no se indique de forma explícita. Cada vez que se quiere añadir o editar una información se tiene que trabajar sobre el proyecto offline.
- Online: Los contenidos que se encuentran en el proyecto Online son aquellos que son visibles en el exterior.

Lo único que requiere para su instalación es un contenedor de JSP/Servlet como Apache Tomcat y un sistema de gestión de base de datos como MySQL.

También admite soluciones propietarias: Microsoft (IIS) y Oracle, como base de datos. A pesar de ser uno de los entornos más estables y depurados de código abierto de los existentes, no tiene un número demasiado elevado de extensiones.

3.1.3 Plone (*de Zope*)

Plone es un sistema de gestión de contenidos basado en Zope [31], concretamente extiende un producto de Zope conocido como CMF (Content Management Framework). Por lo tanto, se debe de tener unas nociones de Zope.

Zope es un servidor de aplicaciones orientado a objetos Open Source. Utiliza el concepto de publicación de objetos y es administrable a través del Web. Un sitio Web de Zope está compuesto por objetos en lugar de por archivos. Lo más característico de Zope es una base de datos orientada a objetos, llamada ZODB o Zope Object Database. Esta base de datos almacena objetos ordenados en un sistema similar a un sistema de ficheros, pero cada

objeto tiene sus propiedades, métodos o contener a su vez a otros objetos. Esta base de datos es muy diferente a las bases de datos relacionales.

Plone está programado en Python. Es un desarrollo basado en código abierto. Puede utilizarse como Intranet, Extranet, un sistema de publicación de contenidos y una herramienta de trabajo de colaboración entre distintas personas/entidades.

Algunas de sus características son:

- Fácil de instalar y utilizar.
- Permite a usuarios no técnicos crear, editar y administrar un sitio Web.
- Prácticamente listo para el desarrollo de contenidos desde la instalación.
- Amplia utilización de estándares como XHTML y CSS.
- Respaldo de una comunidad abierta y muy participativa.
- Solución multiplataforma.

3.1.4 *PostNuke*

PostNuke [4] es un CMS OpenSource y gratuito que deriva de PHP-Nuke y tiene licencia GNU GPL. Está basado en PHP instalado en el servidor, que incluye completas herramientas para construir sitios Webs generados dinámicamente. Destaca su modularidad (cada función viene dada de un módulo distinto) mediante la que se puede incluir en la página Web fuentes RSS, foros, encuestas de opinión, galerías fotográficas, chats, estadísticas, comercio electrónico, buscador interno... Es visible en todos los navegadores y está disponible en varios idiomas. También se puede personalizar su aspecto mediante temas.

El código de PostNuke es orientado a objetos y muy modular. PostNuke guarda la información en una base de datos MySQL. Las últimas versiones permiten usar cualquier base de datos compatible con el controlador ADOdb. PostNuke funciona en cualquier sistema operativo y servidor Web en el que funcione PHP, como Apache o IIS. PostNuke genera código XHTML y CSS válido, y por tanto es visible en todos los navegadores.

Utilizan dos sistemas de caché: pnRender (basado en Smarty) y Xanthia, que permiten que los módulos que sean compatibles con estos sistemas se construyan a partir de plantillas HTML, más fáciles de modificar y que evitan tener que modificar archivos PHP.

Tiene una gran cantidad de archivos, y a veces esto puede complicar su administración.

Se ha cambiado el nombre de este proyecto a Zikula.

3.1.5 *Drupal*

Es un CMS modular y muy configurable [3]. Programa de código abierto con licencia GNU/GPL escrito en PHP, desarrollado y mantenido por una comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas y con un énfasis especial en su usabilidad y consistencia de todo el sistema. Drupal consume más recursos que otros CMS y tiene más módulos de lo que es común usar. Tiene un modulo de autenticación por LDAP.

Algunas de sus características son:

- Ayuda on-line: importante respaldo tanto para usuarios como para administradores.
- Búsqueda: todo el contenido es indexado en tiempo real y se puede consultar en cualquier momento.
- Código abierto: es posible extender y adaptar Drupal según las necesidades.
- Módulos: dispone de una gran cantidad de módulos que aportan funcionalidades como autenticación mediante jabber, mensajes privados, bookmarks...
- Personalización: tanto el contenido como la presentación pueden estar individualizados de acuerdo a las preferencias definidas por el usuario.
- URLs amigables: usa mod_rewrite de Apache para crear URLs que son manejables por los usuarios y motores de búsqueda.
- Autenticación de usuarios.
- Permisos basados en roles.
- Control de versiones.
- Enlaces permanentes.
- Plantillas.
- Independencia de la base de datos

3.1.6 *Mambo*

Es un gestor de contenidos basado en el lenguaje de programación PHP y en la base de datos MySQL de código abierto [5], [15] y [30]. Su principal característica es que permite crear repositorios de información de manera fácil y dinámica, lo cual permite al administrador actualizarla sin que esto suponga un problema. Existe una gran cantidad de información sobre este gestor. Además es efectivo y sencillo en la instalación. Muy intuitivo

tanto para los usuarios como para los administradores, sin necesidad de tener grandes conocimientos técnicos, ni de tener que impartir cursos para su uso. Proporciona un gestor de bases de datos phpMyAdmin, que permite ver rápidamente las tablas que hay en la base de datos, campos de cada tabla, claves de cada tabla y relaciones entre las tablas [10] y [11]. También permite hacer backups de la base de datos de forma sencilla y que en estos backups se almacenen tanto los datos de las tablas propias de mambo, que se crean por defecto al cargar mambo, como los datos de las tablas nuevas creadas por el administrador.

Algunas de sus características son:

1. Es gratis y puede ser usado por cualquier persona, debido a que Mambo es un proyecto que se basa en Open Source.
2. Traducido a numerosos idiomas.
3. Plantillas para modificar el diseño gráfico.
4. Las zonas se pueden personalizar para los usuarios en función de su perfil. Dependiendo del perfil que se le asigne al usuario podrá realizar un tipo de acciones sobre la Intranet.
5. Permite a los usuarios de un cierto perfil incluir sus opiniones sobre los contenidos que se presentan. Así mismo el Administrador tiene forma de modificarlos o eliminarlos.
6. El propio Mambo contiene un motor de búsqueda (tipo Google) ya integrado.
7. Contiene un sistema para generar noticias y mostrarlas vía XML (formato RSS/RDF) automáticamente en otros sitios Web.
8. Contiene unos módulos a través de los cuales se pueden añadir más opciones.
9. Permite distintos niveles de organización de la información.
10. Dispone de una aplicación de encuestas on-line.

Mambo permite integrar lo bueno de todo un diseño Web con un diseño elegante, coherente e integral. Es de los pocos CMS que permite personalizar notablemente la apariencia de la Interfaz.

Mambo ofrece módulos para: Noticias, Canales RSS, Directorio Web, galerías de imágenes, chats, foros, descargas, calendario de eventos, horóscopo, tiendas virtuales...

También ofrece un buen administrador de imágenes, un buen entorno de administración que permite hacer on-line absolutamente todo. También tiene administrador de banners (publicidad on-line), estadísticas, mapa del sitio...

3.1.7 Joomla

Nace a partir de Mambo [13]. La primera versión de Joomla 1.0 es la versión mejorada de Mambo 4.5.3. Evidentemente tiene todas las funcionalidades que había en Mambo y algunas nuevas. Algunas de sus características son:

- Versiones imprimibles de páginas.
- Permite hacer caché de páginas para mejorar el rendimiento.
- Blogs, foros, encuestas...
- Búsquedas en el sitio Web.
- Bookmarks: para directorios de enlaces.

3.1.8 Comparativa entre CMS

CMS	A FAVOR	EN CONTRA
MICROSOFT SHAREPOINT	Integración con todo el paquete Office.	De pago
OPENCMS	Uno de los más estables y depurados.	Basado en Java y Tomcat.
PLONE/ZOPE	Sencillez y gran cantidad de información.	Basado en Python.
POSTNUKE	Estabilidad, rapidez y ayuda on-line.	A igual cantidad de tráfico que Mambo, mayor porcentaje de caídas y desconexiones.
DRUPAL	Gran cantidad de módulos.	Consume más recursos que otros CMSs.
JOOMLA	Versión mejorada de Mambo.	Menos documentación que de Mambo.
MAMBO	CMS que mejor integra un diseño Web de manera elegante, coherente e integral. Gran cantidad de módulos. De los CMS que más permite personificar el FrontPage. Gran cantidad de documentación, ejemplos...	Ya no genera tantos componentes como antes, puesto que de las mejoras se encargó Joomla. Pero para este proyecto concreto, no eran tan necesarios componentes ya elaborados, como información suficiente para poder crear aplicaciones propias para necesidades específicas.

Figura 4: Tabla comparativa de CMS.

Realmente se podría llegar a pensar que la mejor solución posible era Joomla puesto que era la versión mejorada de Mambo, pero en su momento se encontró mayor

documentación de Mambo y realmente con éste se cubrían las necesidades requeridas. No se descartó poder hacer la migración algún día a Joomla.

3.2 Diagramas generales del sistema

Una vez determinada la aplicación y las herramientas para su desarrollo, las funciones que se esperan del programa y los objetivos a alcanzar se inicia la aplicación del modelo.

Como todo desarrollo software, tiene un ciclo de vida que, aunque pueda parecer estático y demasiado general para todas las diferencias existentes entre diferentes aplicaciones software, da una idea general de algunas fases que siempre se van a tener que cubrir.

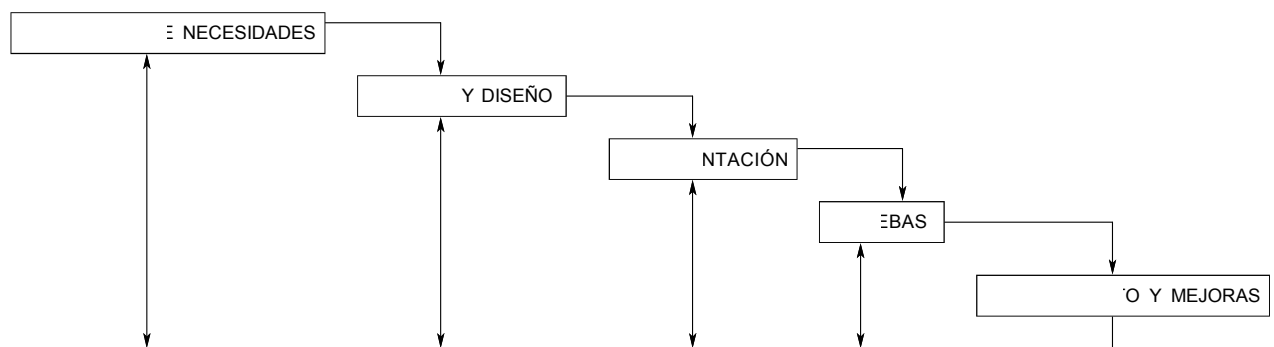


Figura 5: Ciclo de vida software de este proyecto.

Para tener claro que módulos comprende exactamente la Intranet, se va a explicar mediante un diagrama de bloques del sistema. Este diagrama explica desde el soporte base de esta Intranet o la capa más baja de ésta, WAMP con sus propios módulos, hasta los últimos programas realizados con soporte XAJAX. También explica los módulos que contiene cada aplicación y cómo se interrelaciona todos ellos entre sí.

Como se observa en la figura 6, todo parte de WAMP y las aplicaciones que contiene, es decir con WAMP lo que se fija es que el lenguaje de programación a utilizar será PHP, la base de datos MySQL y el servidor Web Apache. Esto es todo lo que WAMP proporciona. Sobre esta arquitectura ya instalada y construida, se pasa a instalar Mambo que es el gestor de contenidos. Mambo va instalado concretamente en la carpeta WWW de WAMP que es la carpeta donde estarán todas las aplicaciones Web. Es decir en la carpeta WWW es donde se

unen WAMP y Mambo. Una vez esté instalado Mambo, a los distintos módulos por defecto y se añadirán para éste sistema en concreto. En la figura 3 los bloques que aparecen en Mambo son los que se han añadido específicamente para este proyecto. En esta figura aparece realmente cómo se han colocado las carpetas con funciones nuevas para este proyecto.

Mosaddphp es un mambot¹ específico que hay que descargarse para permitir que en la página Web aparezca código PHP incrustado en la página sin necesidad de abrir una página nueva. Para que funcione correctamente es necesario tanto el script **“mosaddphp.php”** situado justo debajo de la carpeta *“./mambo/”* como la carpeta *“mosaddphp”* también situada a la misma altura que el script, que contendrá los ficheros PHP que se querrá que aparezcan incrustados en el código. *“template_corporativo”* es lo que hace referencia a los elementos necesarios y como deben estar ordenados para obtener la plantilla.

“Ficheros”, es decir la carpeta denominada *“./ficheros”* situada justo debajo de *“./mambo/”* es la carpeta que se creará para que cualquier usuario pueda subir un fichero en un momento determinado.

¹ Mambot: funciones adicionales de mambo. Fragmento de código que al ser llamados mediante diferentes parámetros pueden activar un programa, un script o ejecutar una función específica.

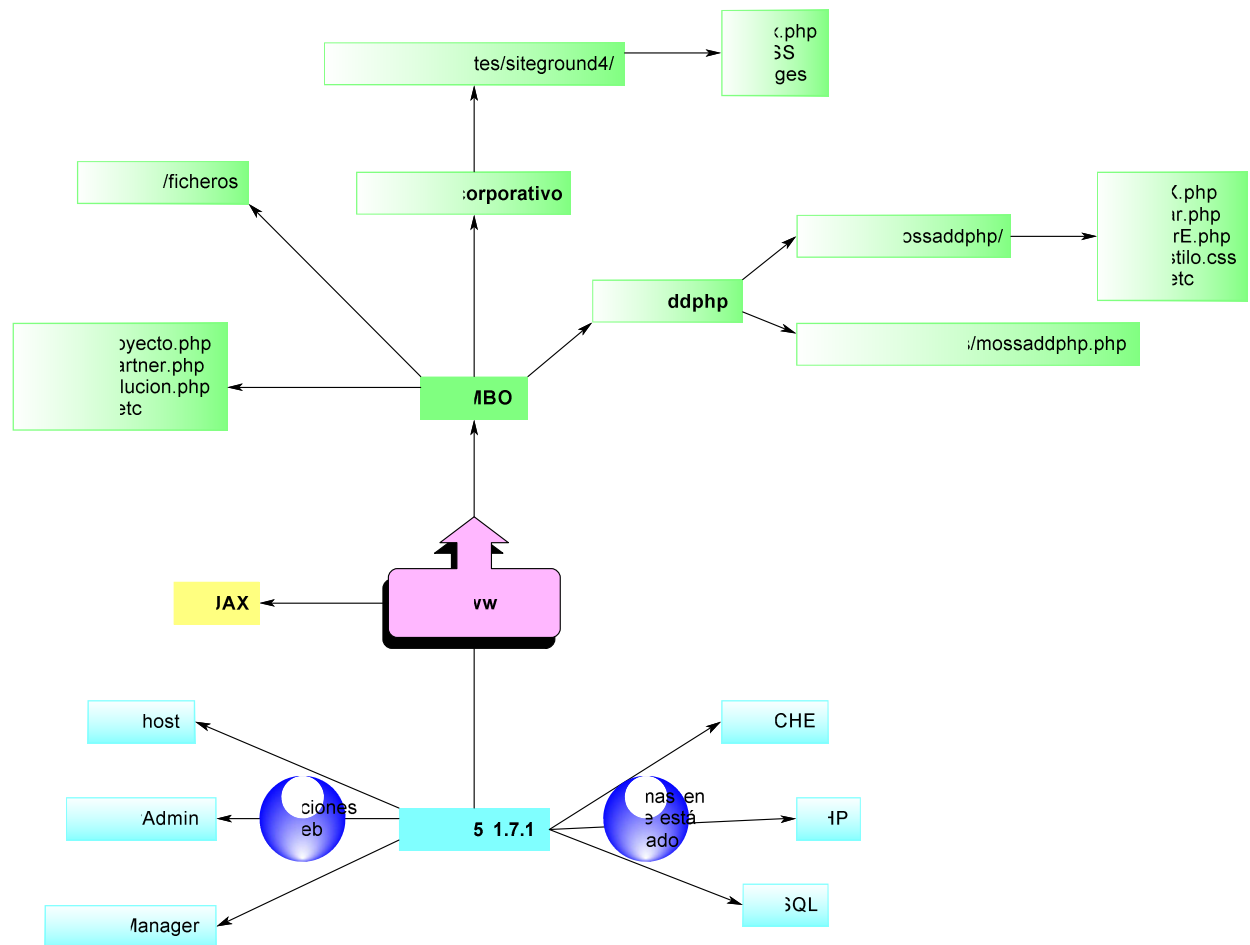


Figura 6: Diagrama de bloques de la Intranet

Los ficheros de extensión que aparecen en este diagrama directamente son aquellos que no van a estar incrustados en la página sino que aparecerán en otra página a parte y que deben estar situados debajo de “./mambo/”.

Xajax debe estar situado justo debajo de WWW.

Esto es tan solo una visión general del diagrama de bloques de la Intranet. Todos estos puntos están desarrollados en el capítulo “4. Implementación”.

3.3 Perfiles de usuarios Mambo

A la hora de dar de alta a los trabajadores como usuarios de la Intranet, cabía la posibilidad de darles de alta con diferentes perfiles. Existen dos grandes grupos de usuarios y estos a su vez tienen dentro diferentes niveles de privilegios.

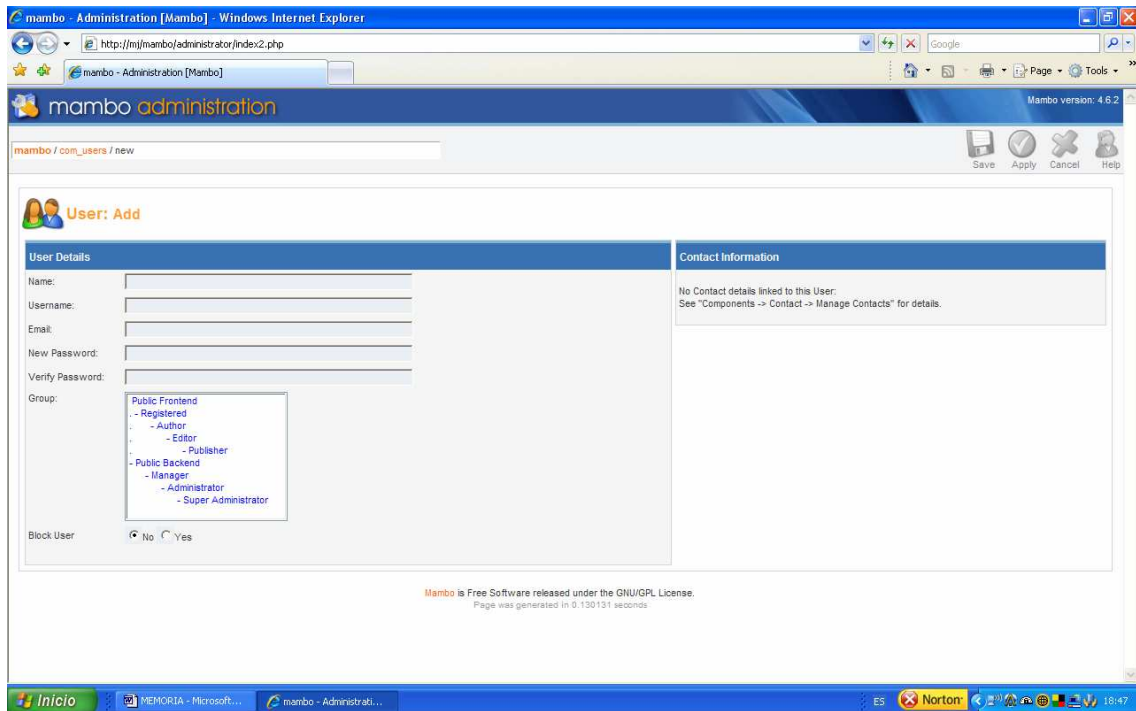


Figura 7: Vista de distintas posibilidades para dar de alta un usuario

3.3.1 *Public Frontend*

Permite a los usuarios la edición, el envío y la publicación de contenidos. El nivel de acceso depende directamente del tipo de cuenta asignada. Para poder acceder a esta zona es indispensable que previamente el administrador haya dado de alta al usuario. En este caso el administrador tiene que proporcionar un nombre de usuario, una contraseña y un tipo de perfil. En este caso para acceder se hace de forma externa a través del formulario que aparece en la Intranet. El tiempo de sesión lo determina el administrador en las opciones generales a las que puede acceder como administrador. Los distintos tipos de usuarios de este tipo son los siguientes (de menor a mayor rango):

- **Registered:** Este grupo es el que menos privilegios tiene dentro de Public Frontend. Solamente se les permite acceder al área de Frontend. Es decir, como de lo que se trata es que a través de la URL de la Intranet solamente se pueda ver la estructura del menú y los titulares de las ultimas noticias, para poder leer todos los contenidos y noticias completas es necesario como mínimo tener el perfil de “Registered”.
- **Author:** Se les permite crear nuevo contenido. Es decir dentro de una categoría concreta cuando aparece el listado con todos los contenidos relacionados, se puede pulsar el botón de “new” y crear así un nuevo contenido pulsando posteriormente el botón “save”. Lo ha creado, pero no lo publica porque no tiene permiso. Sin embargo

el administrador puede ver desde la parte de administración en la lista de todos los contenidos existentes que hay un contenido nuevo que no está publicado y el nombre del usuario que lo ha publicado. El administrador debe valorar si esa información es de interés común, en cuyo caso pasaría a publicarla, simplemente pulsando el botón “Publish”. A los usuarios de este perfil no se les permite modificar contenidos ya creados.

- Editor: Este grupo puede tanto crear nuevo contenido, como modificar el contenido ya existente. Las modificaciones que realice sobre el contenido ya existente no son supervisadas por el administrador. Es decir todos los usuarios con este perfil pueden dejar un comentario, enlace... sobre un contenido sin que intervenga el administrador. Sin embargo siguen sin poder publicar nuevos contenidos sin que sean supervisados por el administrador, al igual que en el caso anterior.
- Publisher: Todo lo anterior y además este permite publicar nuevos contenidos sin tener que ser supervisados por el administrador.

3.3.2 *Public Backend*

Permite crear y editar contenido y publicarlo en un sitio específico de la Intranet, crear niveles y jerarquías, así como administrar las cuentas del tipo Frontend. También permite modificar la apariencia de la Intranet, modificando Templates, imágenes, hojas de estilo CSS, el PHP del “look and feel”... Para acceder a esta zona es necesario conocer la URL del administrador que en este caso es: <http://<servidor>/mambo/administrator/> y tener cuenta con alguno de los perfiles de los que dispone. En definitiva es todo lo que se refiere al administrador pero también con distintos niveles de acceso (de menor a mayor rango):

- Manager: pueden crear contenido y tienen acceso a la información del sistema.
- Administrator: pueden gestionar las cuentas de administrador pero no pueden crear otras cuentas de administrador.
- Super Administrator: se les permite tener acceso a todas las funciones de la administración.

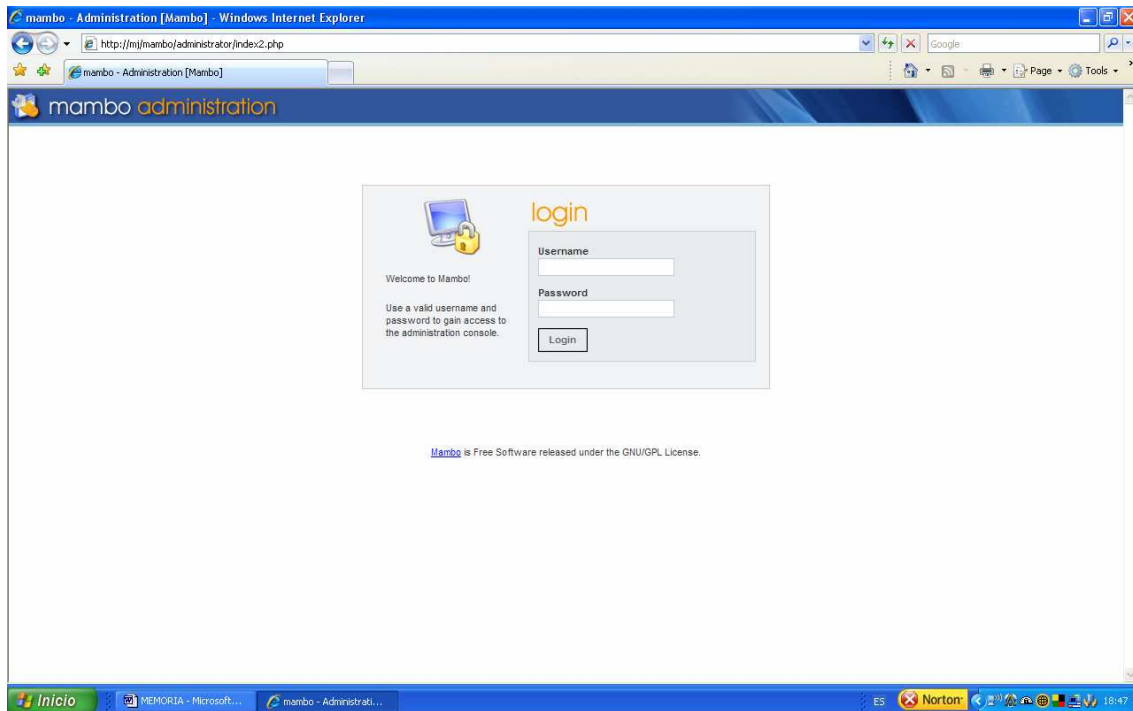


Figura 8: Interfaz de acceso del administrador

En este caso se ha decidido que el encargado del diseño y mantenimiento de la Intranet tenga perfil de “Super Administrator” así como el superior de esta persona, para que al menos dos personas puedan acceder a la parte de administrador. De esta manera la administración de la Intranet recae principalmente en una persona, de ahí su perfil de “Super Administrator”, pero también existe otra persona con este perfil, que también puede trabajar de manera paralela a la encargada principal o actuar de reserva en caso de que hubiese algo que hacer y la persona encargada no estuviese. El resto de compañeros tendrán todos perfiles de Editor. Todos tienen el mismo perfil porque se considera que les proporciona un perfil con los suficientes permisos para que todos los puedan usar.

3.4 Secciones y contenidos

El objetivo del proyecto, es que el sistema que se cree sea capaz de manejar una cierta cantidad de contenidos, los cuales deberán de estar organizados de acuerdo a algún criterio. Para ello, se definirá un esquema de presentación de los contenidos, lo más intuitivo posible para el usuario. Este esquema se deberá de ver desde el menú principal de la Intranet e irá ordenado de mayor a menor grado en secciones, categorías y subcategorías. Por ejemplo, será necesario crear una sección denominada **OTROS DEPARTAMENTOS** que hará referencia a otros departamentos con lo que se relaciona el departamento donde la Intranet

funciona. Departamentos, como por ejemplo compras, marketing, operaciones... Después en cada uno de estas categorías irán los contenidos relativos a ellas. Es decir, dentro de cada una de estas categorías habrá uno o varios contenidos con información relativa a ellas. Si fuese necesario, también se podrán crear subcategorías y categorías a más bajo nivel.

3.5 Base de datos de conocimiento y soluciones externas

Para la completa realización del proyecto, será necesario crear una base de datos dónde poder almacenar nueva información o información que se tenía pero no estaba demasiado bien organizada. Esta base de datos se creará al mismo nivel que la base que por defecto ya se crea al cargar el gestor de contenidos. La base de datos debe ocuparse de tres partes concretas: soluciones externas, proyectos y estudios de viabilidad. Cada una de estas partes necesitará una serie de tablas asociadas y cada una de estas tablas tendrá los registros correspondientes. A la hora de representar esta información en la Intranet, se hará un inciso especial con una sección concreta al apartado de **SOLUCIONES EXTERNAS**, donde habrá una categoría por cada una de las soluciones más importantes o que los usuarios van a consultar más a menudo.

3.6 Conclusiones del capítulo

Al finalizar esta parte se tienen todas las herramientas necesarias para desarrollar la plataforma de repositorio de información. Contando con el análisis que llevó a decidir la tecnología a utilizar, información suficiente sobre todas las tecnologías involucradas en el proyecto, se ve claro los requisitos que debe cumplir la Intranet.

El siguiente paso es la instalación y configuración de los elementos necesarios para el funcionamiento de la Intranet.

En el apéndice aparece el diagrama de clases PHP, ya que hay en este proyecto intervienen un número importante de clases PHP con nombres y funciones que pueden parecer similares y para que no haya lugar a duda se inserta el diagrama completo con todas las clases PHP que se han creado.

4 Implementación

Para la puesta en marcha de este proyecto es importante tener en cuenta antes de empezar todos los elementos (scripts, ejecutables...) imprescindibles para poner la Intranet en funcionamiento.

- Ejecutable de Wamp5 1.7.1. Es la base de la que parte toda la Intranet, sobre la cuál se construirá todo lo demás. Es el servidor Web que alojará la Intranet.
- Ejecutable de Mambo. CMS que estructurará la información que se desea publicar en la Intranet. Una vez que ya se tiene un servidor Web dónde poder publicar páginas, con Mambo ya se tendrá la Intranet para publicar pero faltará personalizarla.
- Plantilla corporativa. Una vez que se tiene el administrador de contenidos funcionando y la estructura creada, debe tener la imagen corporativa. Para ello, se necesita una plantilla que se pueda adaptar fácilmente a la imagen que se quiere obtener.
- Librería XAJAX. Con toda la Intranet ya funcionando, se pasa a añadir funcionalidades para mejorar el funcionamiento.

La instalación completa de la Intranet es compleja y larga. Tiene diferentes aspectos a tratar y cada uno de ellos se trabaja en una fase distinta de la instalación. Por este motivo, el desarrollo completo de la Intranet se explica por fases.

Se comienza con la instalación de la parte más básica, y se va desarrollando hasta terminar por depurar detalles.

4.1 Instalación y obtención básica del sistema.

1. Instalar Wamp5 1.7.1.

Para ello se pueden seguir las instrucciones que aparecen en páginas Web como por ejemplo:

<http://www.desarrolloweb.com/articulos/wampserver-instalar-php5.html>

De esta parte de instalación, es importante cambiar el parámetro de localhost por el nombre de la máquina ² para que pueda acceder (con localhost solo se puede acceder desde la propia máquina).

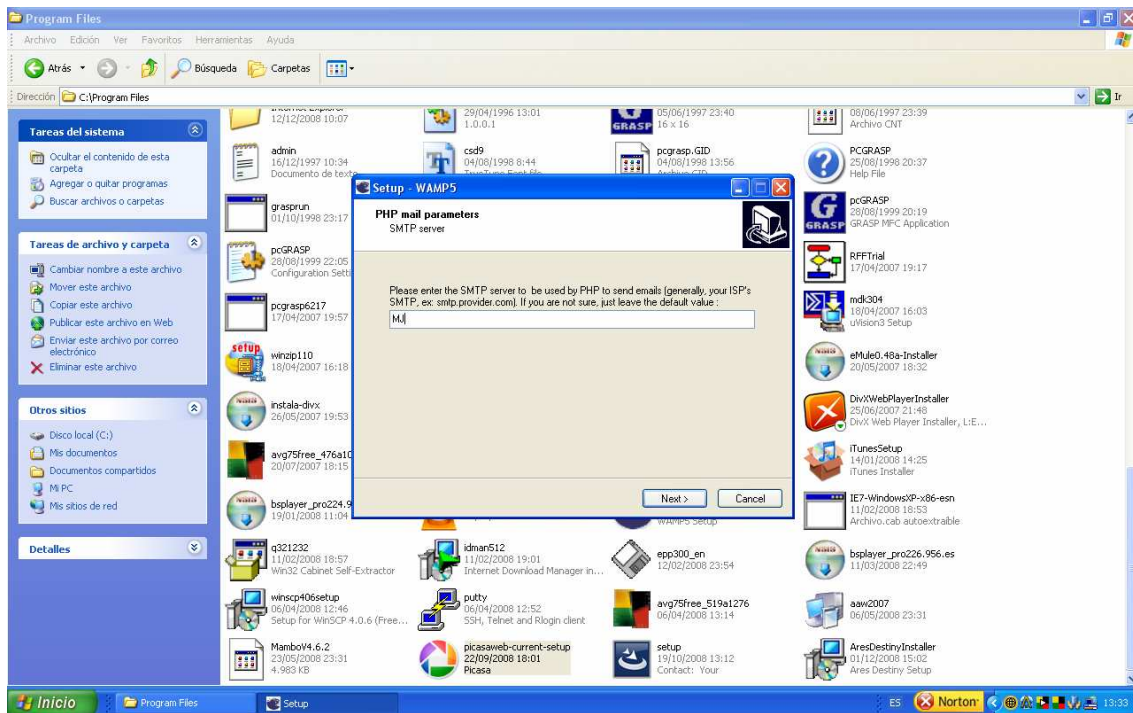


Figura 9: Instalación de Wamp

Otro de los puntos a destacar es la selección del documento raíz del sitio Web (DocumentRoot), es decir la ruta para poder acceder a la interfaz de WAMP. Dependiendo del valor escogido, después de la instalación al teclear **http://localhost/** aparecerá una u otra cosa. Si por ejemplo se escoge WAMP como DocumentRoot al acceder a través de la consola de Wamp a Localhost aparecerá algo como:

² En este caso, la máquina de desarrollo se llama “mj”.

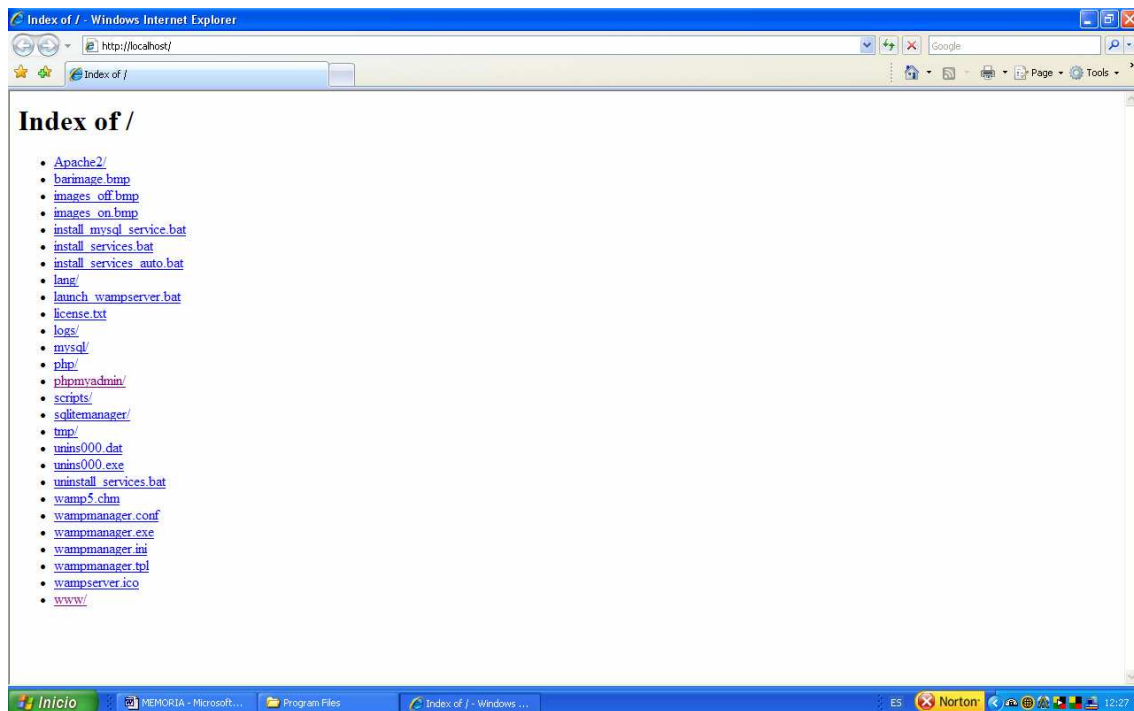


Figura 10: Vista de localhost

Desde aquí, para llegar a la interfaz propia de Wamp habría que pulsar WWW, entonces se obtendría la página deseada, la mostrada en la figura siguiente:

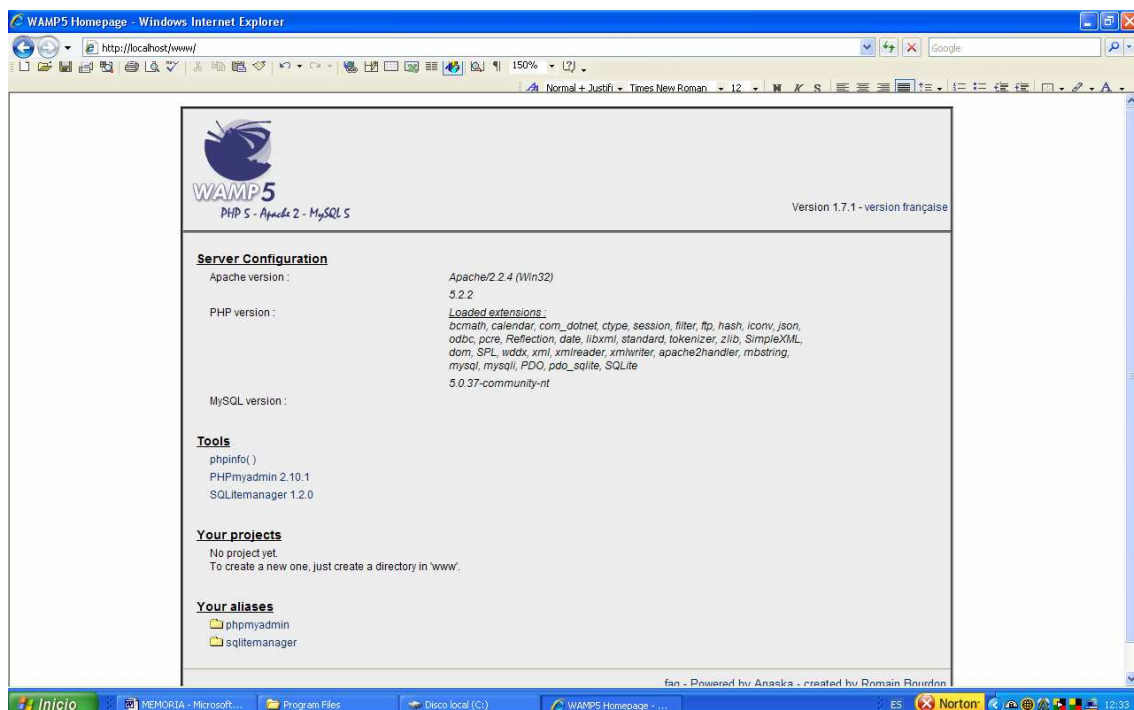


Figura 11: Vista de Wamp instalado

Sin embargo, si como DocumentRoot se escoge WWW entonces al volver a realizar la misma operación se obtendría la misma pantalla pero con un paso menos.

Es indiferente escoger uno u otro DocumentRoot, pero sí es importante tenerlo en cuenta para saber cómo se estructura el sistema de directorios y carpetas.

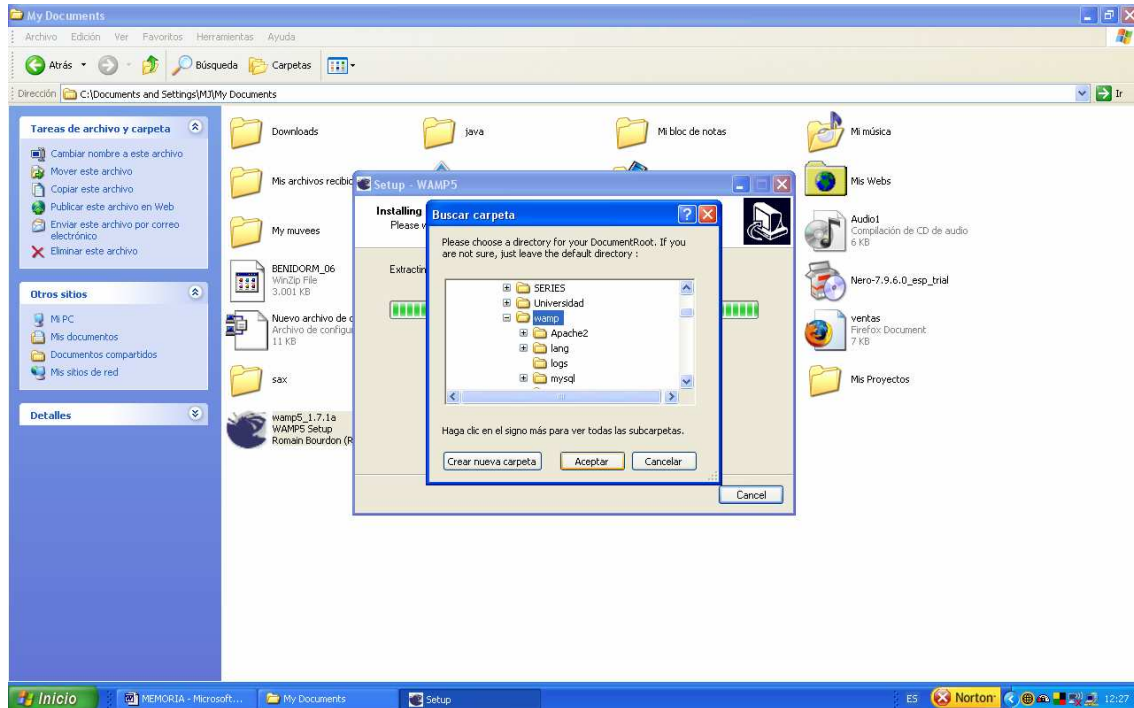


Figura 12: Vista de sistema de carpetas en Wamp

Se comprobará que se habrá instalado correctamente cuando el semicírculo inferior derecho esté completamente blanco. Otras maneras de saber si se ha instalado correctamente es mirando en el navegador en el propio localhost, es decir, tecleando en el navegador `http://localhost/` se debería obtener una imagen similar a la siguiente figura.



Figura 13: Vista de sistemas de directorios existentes en localhost

Se puede ver la presentación Web de WAMP con todas las aplicaciones incluidas. En la sección de “Your projects” irán los proyectos que crea el usuario, es decir será ahí donde irá el CMS, por eso de momento no aparece nada. Otra comprobación de que la instalación ha sido correcta es intentar acceder a la misma interfaz en lugar de poniendo en la URL “localhost” poniendo el nombre de la máquina, en este caso “mj”, obteniéndose una página como la de la figura.



Figura 14: Comprobación de instalación válida

- Una vez que Wamp5 1.7.1 está funcionando correctamente, hay que copiar el instalable de Mambo en “./wamp/WWW/”. Descomprimir todo en una sola carpeta y cambiarle el nombre, por ejemplo a ‘mambo’ y borrar el instalable anterior. A partir de aquí se mantendrá siempre ese nombre. Accediendo a WWW ya se puede ver que aparece “mambo” en la sección “My projects”.

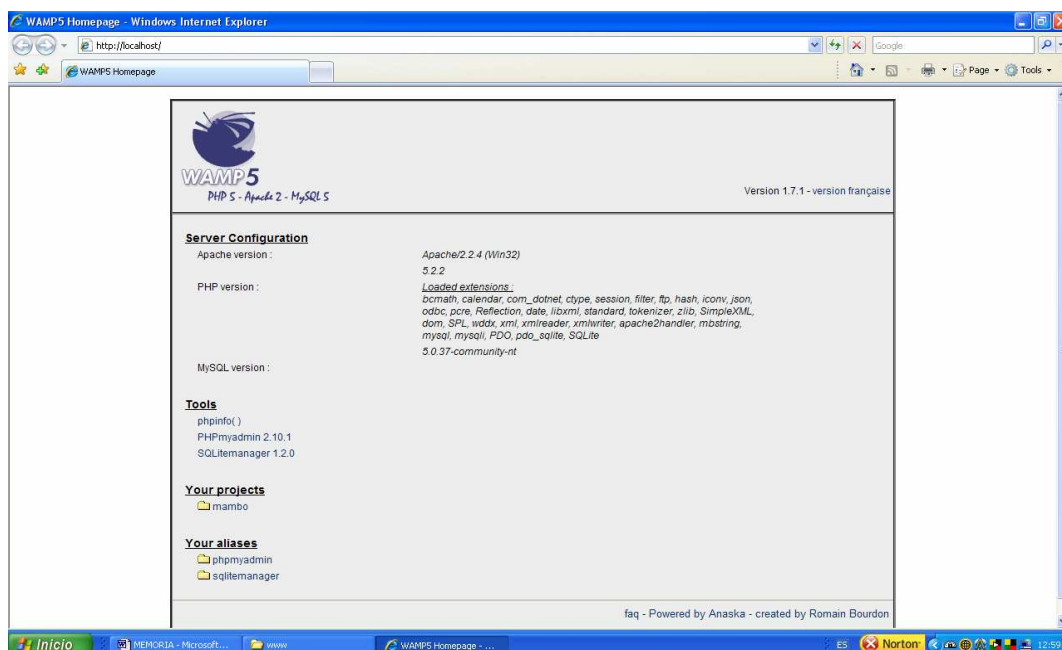


Figura 15: Vista de WWW una vez está Mambo instalado

3. Ahora mambo ya está ubicado donde le corresponde pero no está instalado. Para instalarlo hay que acceder al icono de WAMP, después pulsar localhost (o “mj”) y ‘mambo’ o el nombre que se le haya puesto al descomprimir la carpeta. Al acceder aquí se accede a las páginas de instalación de Mambo. Se deben seguir los pasos indicados para la instalación. Para esto se pueden seguir los pasos del siguiente enlace (aunque en Google aparecen numerosas páginas explicando este proceso):

<http://www.cmsjoomla.com.ar/mambo/instalacion-de-mambo.html>

De la instalación de Mambo una de las cosas más importantes es la página de instalación que te pide el nombre del sitio, en el paso 2. Aquí hay que poner “mambo” . Realmente se podría poner cualquier nombre, pero será conveniente poner un nombre fácilmente reconocible.

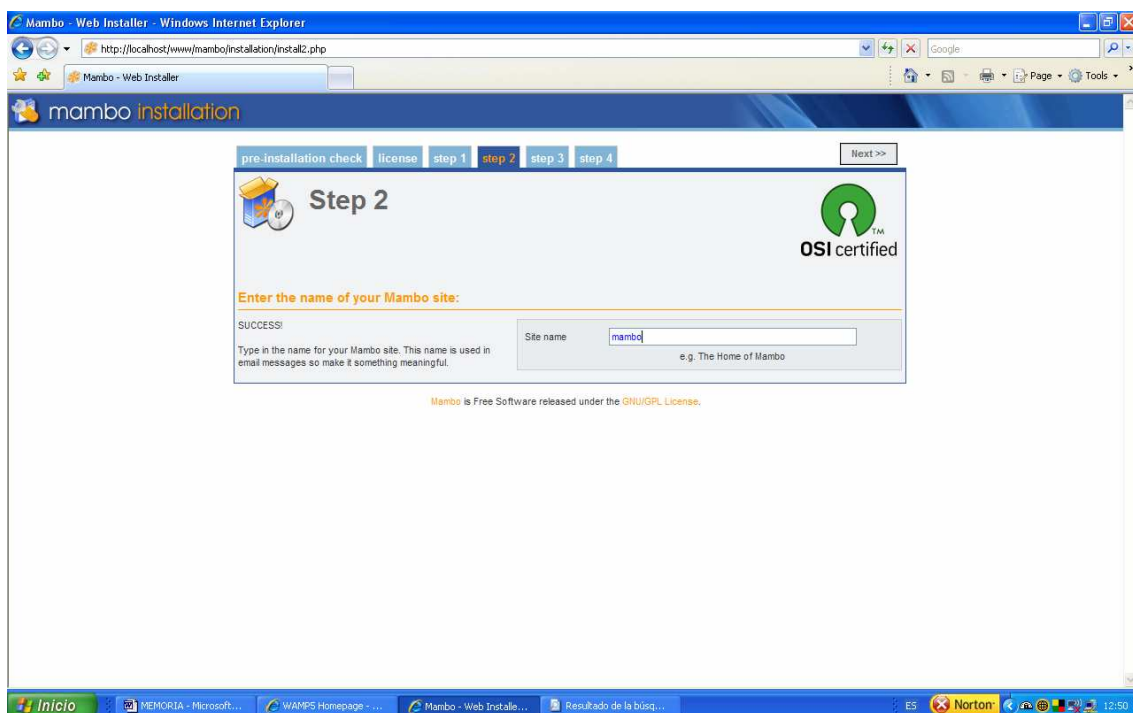


Figura 16: Paso 2 de la instalación de mambo

En la penúltima hoja de instalación aparece la URL definitiva, por defecto aparece (en caso de haber puesto como nombre “mambo”):

<http://localhost/mambo/>

Es muy importante cambiar esta URL por esta otra, con el nombre de la máquina:

http://mj/mambo/

Es mejor el nombre de la máquina porque por la IP o localhost hay aplicaciones que no lo reconocen.

En los ordenadores que no son el servidor, si para acceder a la Intranet lo tienen que hacer a través de **http://localhost/mambo/** no podrán resolver “localhost” y por tanto, no podrán ver nada. Este es el motivo por el cuál este paso es importante, sin él no se puede acceder a la Intranet desde fuera del servidor.

Por otra parte en esta página también aparece la clave de administrador.

Una vez que haya finalizado la instalación de Mambo se borra la carpeta ‘installation’ que está en “mambo” por seguridad. Esto también lo indican las instrucciones por pantalla.

4. Ahora, al acceder a la URL de mambo, se debería ver en todas las máquinas la plantilla y la base de datos que carga por defecto, según se muestra en la siguiente figura.

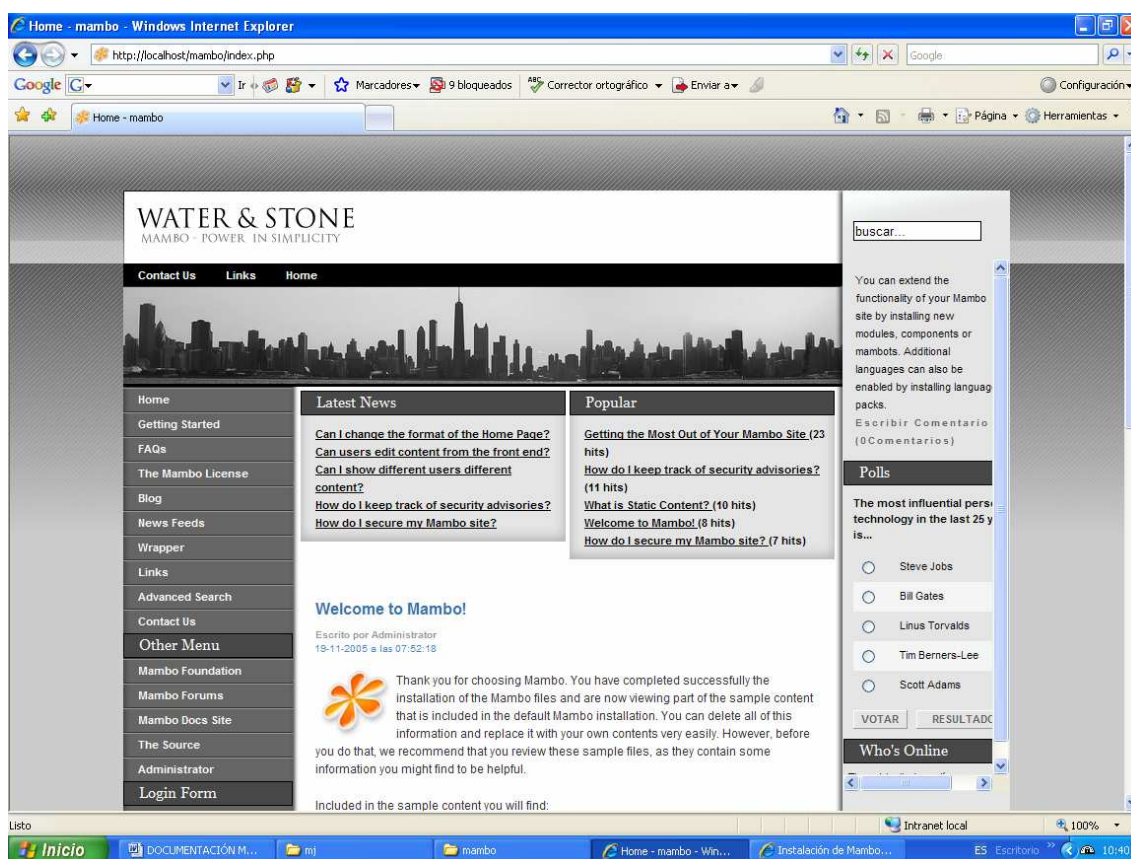


Figura 17: Vista de Mambo instalado con valores por defecto.

4.2 Configuración de la imagen corporativa.

Por lo tanto ahora para obtener la Intranet que se desea quedan varias cosas por hacer que el Template sea lo más corporativo posible [17], se instalará un nuevo Template que se asemeje a lo que se desea y si es necesario se modificará el código para ajustarlo a dicha imagen. Evidentemente, cuánto mas se asemeje a la apariencia que se desea menos modificaciones habrá que hacerse en la hoja de estilo y código HTML.

Se puede escoger un Template de los que hay por ejemplo en la página:

<http://www.siteground.com/mambo-hosting/mambo-templates.htm>

En este caso se ha utilizado el que se denomina NUsite y su mayor aplicación se da en IT Business y la carpeta que se descarga tiene como nombre siteground4.

Una vez descargado, se coloca la carpeta descomprimida en “./mambo/Templates”. A la hora de trabajar con los templates es importante que sea la carpeta “wamp\www\mambo\templates” y no en la carpeta “wamp\www\mambo\administrator\templates” ya que a pesar de tener los mismos nombres solo se pueden modificar y gestionar los templates de la primera.

Una vez esta carpeta está copiada se accede a mambo como administrador

<http://mj/mambo/administrator>

y en **Sitio>Administrador de Plantillas> Plantillas del sitio** se selecciona el nombre del Template que se ha descargado y se pulsa el icono de “Por Defecto”. Los pasos a seguir a continuación serán los siguientes:

- i. Una vez el template tenga el aspecto deseado se creará la estructura del menú con la información que más interesa.
- ii. Con el Template y el menú de contenidos en funcionamiento, se pasará a crear la base de datos con los datos de lo que dispone la empresa relativos a EVs, Proyectos y Soluciones.
- iii. Habrá que programar los PHP necesarios para que el usuario pueda acceder a la información de la base de datos desde la Intranet.
- iv. En algunos de los archivos PHP será necesario el uso de XAJAX, por lo tanto habrá que instalar los scripts y ejecutables necesarios para el funcionamiento de XAJAX.

Ahora al acceder a la URL debería tener la apariencia del Template escogido aunque la información corresponda con la anterior. La imagen que se debería ver sería algo como la mostrada en la figura siguiente.

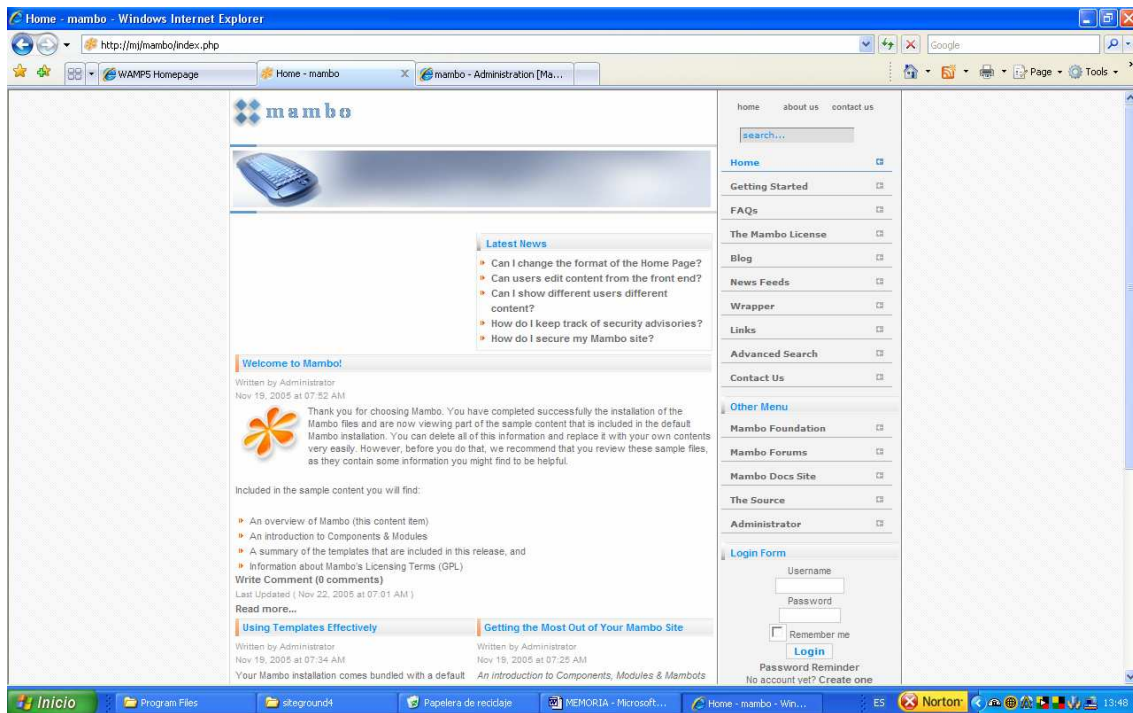


Figura 18: Vista del Template de Mambo que se va a usar

De esa imagen que se ha obtenido se quiere cambiar los siguientes detalles:

- El contenido de la parte principal de la página debe aparecer en una sola columna.

Para ello, se accede a la interfaz de administrador y de ahí se va a **Menú → MainMenu** y entonces aparece el listado de todas las categorías que existen. Ahora se entra en todas las categorías y en “Parameters” se comprueba que el número de columnas sea 1. Si no es 1, se modifica y se pulsa “Save”.

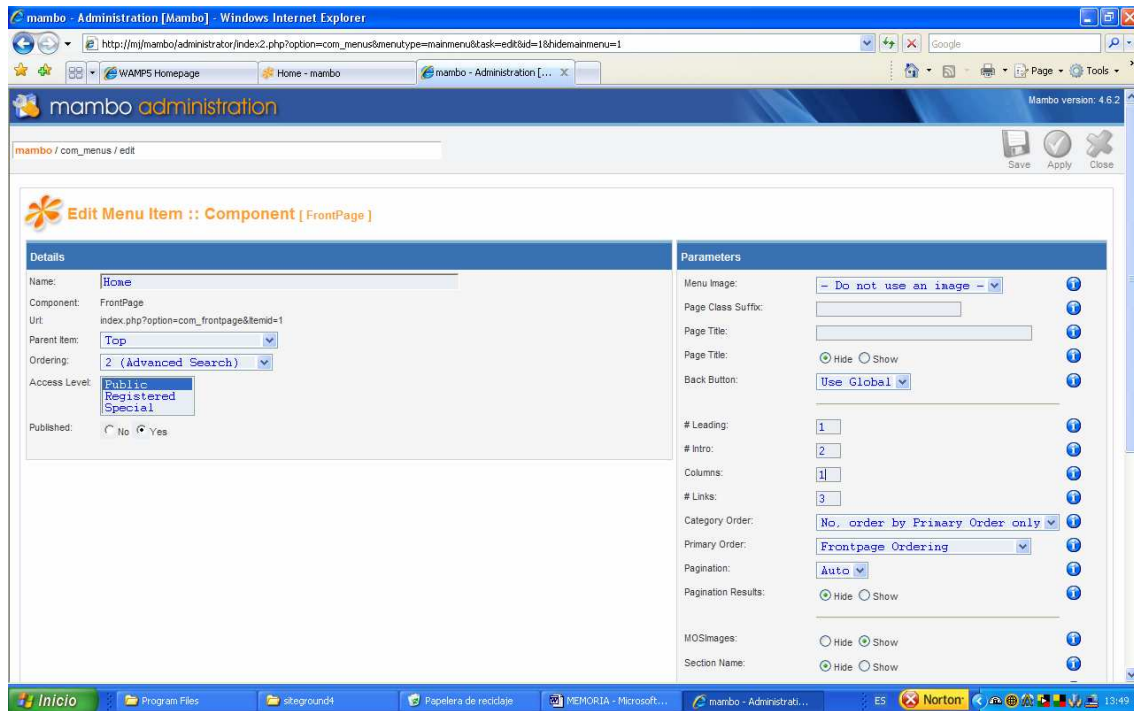


Figura 19: Vista de la edición del FrontPage.

Con lo anterior ya se obtiene que los contenidos de la página principal aparezcan ordenadamente uno debajo del otro.

- Que desaparezca en el menú de la derecha la parte inferior donde pone “Syndicate”. Se accede a la interfaz del administrador **Modules> Site Modules**, se localiza **Syndicate** y en la columna de Publisher que indica que módulos aparecen y cuáles no, se pulsa para que este modulo no sea publicado.
- Que la cabecera de la página sea la corporativa. Primero se crea la propia cabecera corporativa. Una vez que se tiene la cabecera que se desea poner, se tiene que guardar en la carpeta de imágenes del Template, es decir en **siteground4>images**. En este caso la imagen se denomina “**business.gif**”. Ahora se sustituye la que había por defecto por la que se acaba de crear. Para ello menú de administrador y se accede a **Site> Template Manager > Site Templates**. Se selecciona el Template que se esta usando y se pulsa “Edit HTML”. Ahora aparece todo el código HTML de la página principal. La sustitución que hay que hacer es donde antes ponía:

```

<div id="logo_bg"><div id="sitename"><?php echo
$GLOBALS['mosConfig_sitename']?></div></div>

<div class="clr"></div>

<div id="hr"></div>

<div id="banner"></div>

<div id="hr"></div>

```

ahora hay que poner:

```

<div id="logo_bg"></div>
</div>

```

Y el resultado obtenido es el mostrado en la figura siguiente:

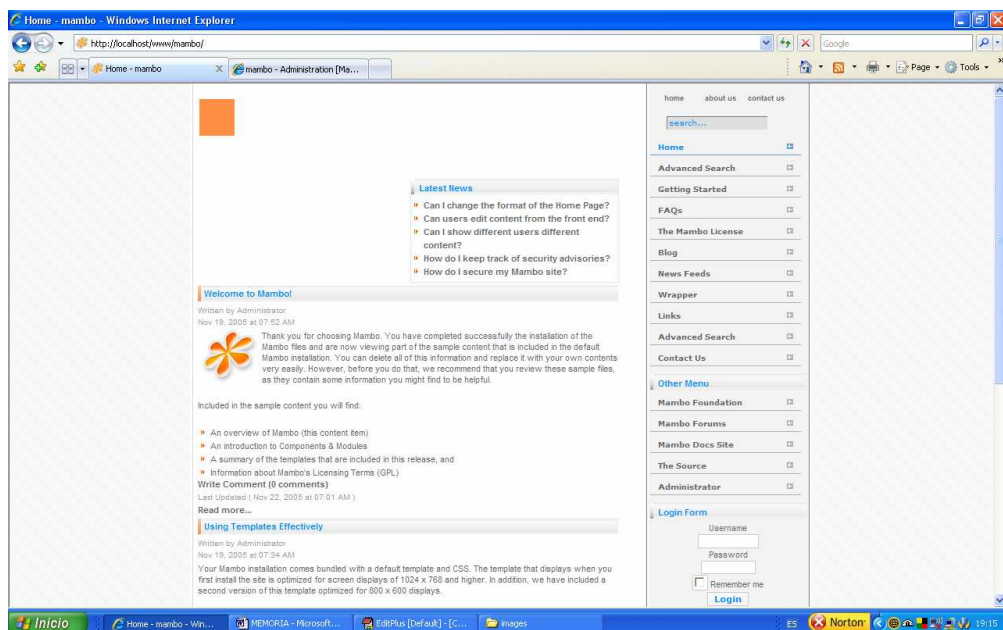
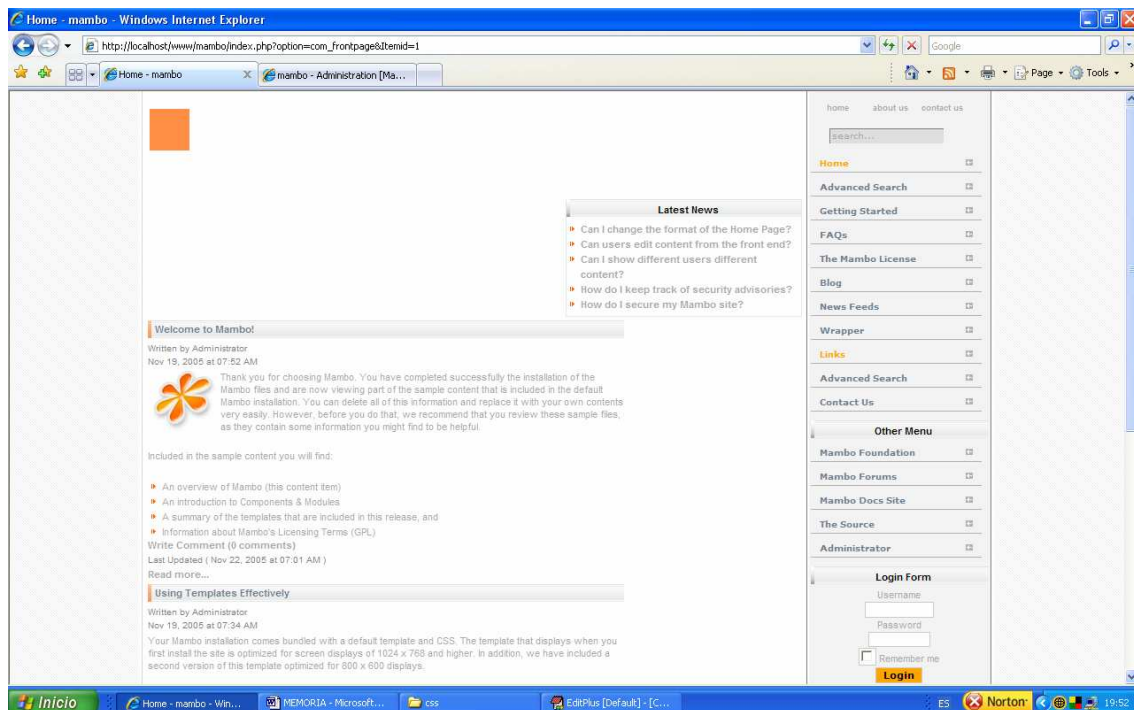


Figura 20: Vista de Mambo con algunos cambios realizados.

Que las partes que hasta el momento aparecen subrayadas en azul, aparezcan en naranja y que al pasar el cursor por encima del menú derecho, por donde vaya pasando en lugar de quedarse en azul se quede en naranja. Para ambos casos las modificaciones se deben hacer en la hoja de estilo CSS. Para ello se accede al interfaz del administrador hasta llegar al listado de Templates y allí se selecciona el Template y se marca la opción “Edit CSS”.

La página que se obtiene es la mostrada en la siguiente figura.

**Figura 21: Vista de Mambo con la CSS modificada.**

- Que el módulo de “Latest news” no ocupe sólo la mitad del espacio de la parte frontal de la página, sino el espacio completo. Además, que en lugar de aparecer únicamente las 5 últimas noticias o modificaciones hechas en la Intranet, aparezcan 10 noticias. Para aumentar el número de noticias que aparece en módulo de “Latest News”, hay que acceder al panel de

administrador **Modules**> **Site Modules** > **Latest news** > **Parameters** > **Count** ahí donde pone 5 escribir 10 y guardar los cambios. Para que el módulo aparezca en el centro de la parte central de la página deben hacerse dos cosas. Primero acceder al módulo y en la opción de **Details**> **Position** seleccionar “user3” y guardar los cambios. Conseguido esto, ahora hay que modificar el código HTML del Template. Hay que sustituir el código:

```

<?php
    if ( mosCountModules( 'user1' ) ) {
        ?>
        <div id="user1">
            <?php mosLoadModules ( 'user1' ); ?>
        </div>
    <?php
    }
    if (mosCountModules( 'user2' )) {
        ?>
        <div id="user2">
            <?php mosLoadModules ( 'user2' ); ?>
        </div>
    <?php
    }
    ?></div>

<?php mosMainBody(); ?>
<div class="clr"></div>

```

por el código:

```

<table height="100%" width="100%" align="center">
    <?php mosLoadModules ( 'user3' );?> </div>
<div class="clr"></div>

```

5. Ahora ya debería aparecer en el centro. Puede darse el caso de que además de aparecer el módulo que se quiere en el centro aparezca otro, como por ejemplo “Popular”. Esto es debido a que en el cambio del HTML se ha indicado que lo que debe aparecer al principio son los módulos cuya posición sea “user3”. Puede darse el caso que haya otros módulos con esta posición. Para solucionarlo se pueden cambiar la posición de los módulos que también tengan posición “user3” o si como es este caso no interesan otros módulos, dejarlo de publicar. Con esto ya se obtiene la vista de la siguiente figura.

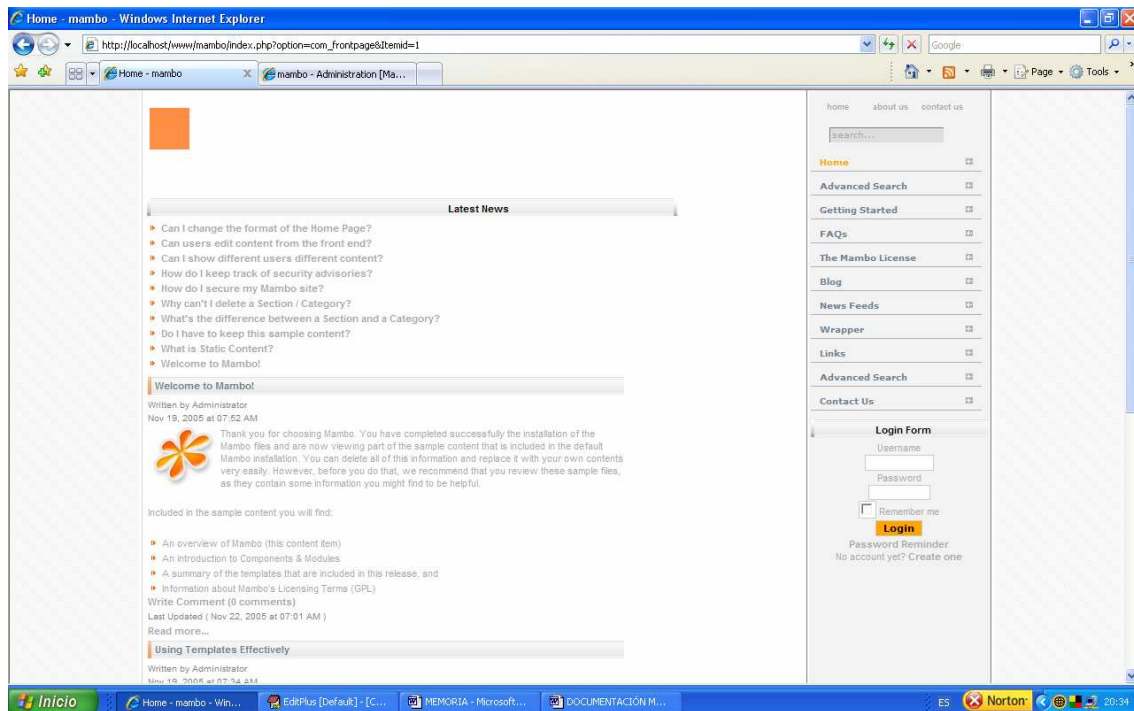


Figura 22: Vista de Mambo con el Template definitivo.

4.3 Creación de estructuras y contenidos.

Una vez que se tiene la imagen deseada se va a determinar la estructura que tendrá el menú. Los contenidos en Mambo se clasifican (de mayor a menor grado) en secciones, categorías y contenidos. Como previamente ya se ha decidido qué elementos van a ser las secciones, qué categorías van a contener cada sección y que contenidos van a corresponder a cada categoría para comenzar, puesto que esto después se podrá modificar, solamente queda llevarlo a cabo, es decir crear la estructura. Para ello se detalla a continuación cómo se va a llevar a cabo la creación de la categoría “OTROS DEPARTAMENTOS” con sus correspondientes secciones y éstas a su vez con sus correspondientes contenidos. El resto de estructura del menú se hará de forma análoga a esta sección.

1. Primero se accede a la aplicación desde la Interfaz de administrador. Se va al panel superior y se pulsa **Content > Section Manager**. Ahí se ve una pantalla con la lista de Secciones que existen con sus datos como nombre, si esa sección está publicada o no, cuantas categorías contiene... Para crear una nueva sección se pulsa el icono de “New”. Ahí solo hay que completar los campos “Title” y “Section Name”,

escribiendo en ambos lo mismo “OTROS DEPARTAMENTOS”. Se pulsa el icono de “Save” para que este cambio quede guardado y se vuelve a la pantalla del listado de secciones existentes. Hasta ahora solo se ha indicado el nombre de la sección, pero hay secciones de varios tipos y hay que indicar cuál es el adecuado. Para ello se busca la sección que se acaba de crear en la lista de secciones y se pulsa encima de su nombre. Ahora se ve a la derecha de la pantalla un apartado llamado “Link to Menu”. Ahí se tendrá que completar lo siguiente:

- i. **Select a menú→ mainmenu.** Esto es porque se quiere que esta sección aparezca en el menú principal y no solo en el menú del usuario logado.
 - ii. **Select menú type→ Section List.** Con este tipo se consigue la apariencia deseada.
 - iii. **Menu Item Name→OTROS DEPARTAMENTOS.**
 - iv. Se pulsa el botón de “Link to Menu” y se vuelve a guardar las modificaciones echas en la sección.
2. Con esto ya se ha conseguido que en el menú aparezca el apartado “OTROS DEPARTAMENTOS”, pero si se pulsa sobre él, no aparece nada porque todavía no se le han añadido categorías. Este hecho se puede ver en la siguiente figura.

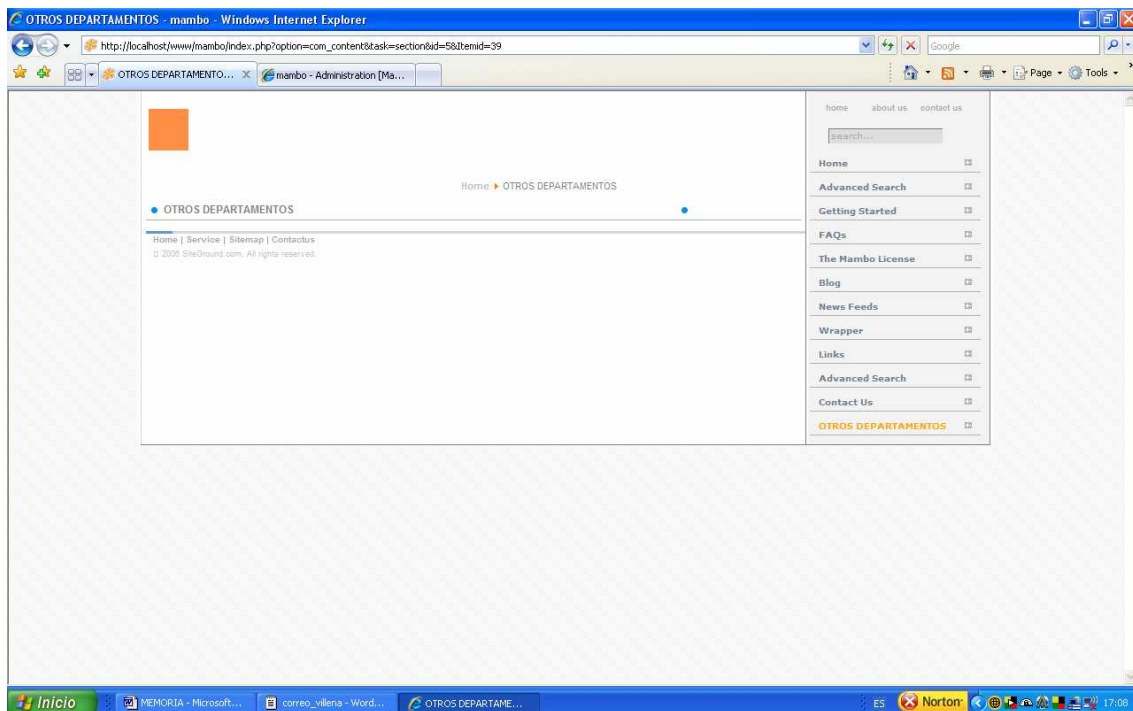


Figura 23: Vista de una sección sin categorías.

Se quiere que al pulsar sobre él aparezca un desplegable con seis categorías. Para añadir categorías a una sección ya creada en modo de desplegable habrá que hacer lo siguiente:

- Dentro del menú del administrador hay que acceder a **Content> Category Manager** allí aparecerá un listado de categorías con sus características y se pulsa el icono de “New”. Se debe completar los campos de “Category Title” y “Category Name”, en este caso por ejemplo se escribe “COMPRAS”.
- Después viene el apartado “Section” dónde hay que seleccionar en el desplegable la sección que antes se ha creado y se pulsa “Save” volviendo así al administrador de categorías. De momento si se vuelve a pulsar sobre “OTROS DEPARTAMENTOS” siguen sin aparecer las categorías.
- Para que la categoría aparezca, hay que volver a acceder hasta la categoría que se desea que aparezca. Hay que hacer algo similar a lo que se ha hecho con la sección. En “Link to Menu” se selecciona “mainmenu” y después “Select Menu Type” se selecciona “Content Category Table” y en “Menu Item Name” se escribe el mismo nombre que el de la categoría, en este caso “COMPRAS” y se pulsa “Save”.
- Ahora en el menú principal aparece la categoría “COMPRAS” como si fuese una sección, es decir al mismo nivel que “OTROS DEPARTAMENTOS” cuando lo que se quiere es que “COMPRAS” este dentro de la sección en lugar de a la misma altura. Para conseguir esto, hay que colocarse en **Menu> Mainmenu > COMPRAS** y hay que dejar seleccionadas dos cosas. Por un lado donde pone “Category” hay que seleccionar “OTROS DEPARTAMENTOS / COMPRAS” con esto se consigue que cuando se pulse en “COMPRAS” solo se vea la información relativa a “COMPRAS” y no a otras secciones o categorías. En Parent Item hay que seleccionar “OTROS DEPARTAMENTOS”, que es dónde se dice debajo de quién va a aparecer en el menú de presentación. Se guardan los cambios y ahora cuando se pulse en “OTROS DEPARTAMENTOS” ya aparecerá en un nivel inferior la opción “COMPRAS”.
- Todavía falta a esa categoría “COMPRAS” añadirla contenidos. Para ello hay que ir a **Content > Content by section > OTROS DEPARTAMENTOS > NEW**. Ahora en “Title” y “Title Alias” se escribe lo mismo, es decir “CONTENIDO 1”. En el lado derecho, en “Section” debe poner “OTROS DEPARTAMENTOS” y en “Category” debe poner “COMPRAS”. Y abajo en la zona que simula al Word habitual se escribe

“PRUEBA CONTENIDO 1” y se prueba a darle un formato por ejemplo centrarlo, ponerlo en negrita, cambiarlo a color negro y a un tamaño de 6.

- Almacenar los cambios. Ahora si a través del menú del usuario se accede a **OTROS DEPARTAMENTOS > COMPRAS** se verá algo como esto:

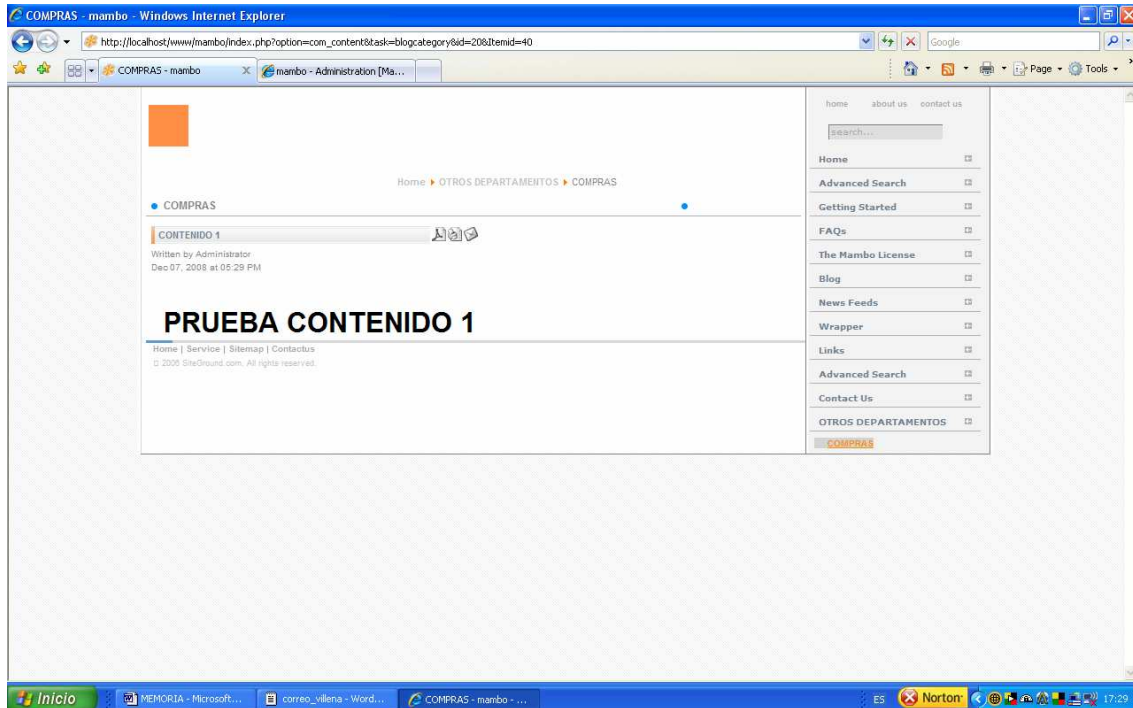


Figura 24: Vista de un contenido.

3. Ahora hay que modificar algunos detalles de diseño. Por ejemplo, que al ver un contenido no aparezcan los iconos de PDF, Print y correo-e puesto que no se consideran necesarios para el uso que se le va a dar a esta herramienta. Para que no se muestren estos iconos hay que ir a la interfaz de administrador **Site > Global Configuration > Content** y ahí se busca en la lista **PDF Icon**, **Print Icon** y **Email Icon** y se cambia su situación de “Show” a “Hide”. Se almacenan los cambios.
4. Por otro lado, de momento siguen apareciendo las secciones que venían por defecto más las que han sido añadidas como nuevas. Solo se quiere que aparezca “HOME” y el nuevo esquema de secciones y categorías de la información. Por tanto hay que ir a **Menu > MainMenu** y marcar como no publicadas todas aquellas que no correspondan con HOME o las nuevas secciones. Tras ello, quedaría la página principal algo como lo que se muestra en la figura siguiente.

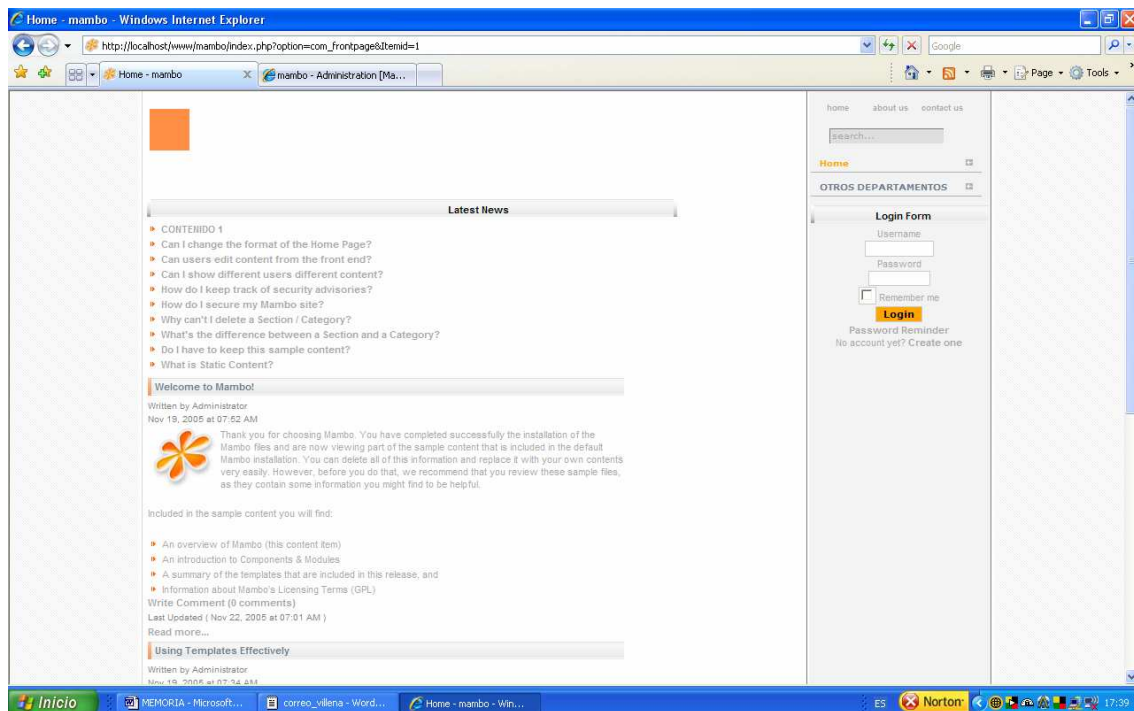


Figura 25: Vista de proceso de creación del Template.

5. El menú ya está solo con la nueva información, pero sigue habiendo contenidos que no son los nuevos en la parte principal. Se va a la interfaz de administrador **Content > All Content Items** y se marcan como no publicados todos los contenidos excepto “CONTENIDO 1” que es el nuevo contenido de prueba. También hay que entrar en **Content > Static Content Manager** y marcar como no publicado todos aquellos contenidos que no sean nuevos. Con estos cambios, ahora la página principal sería similar a la siguiente figura.

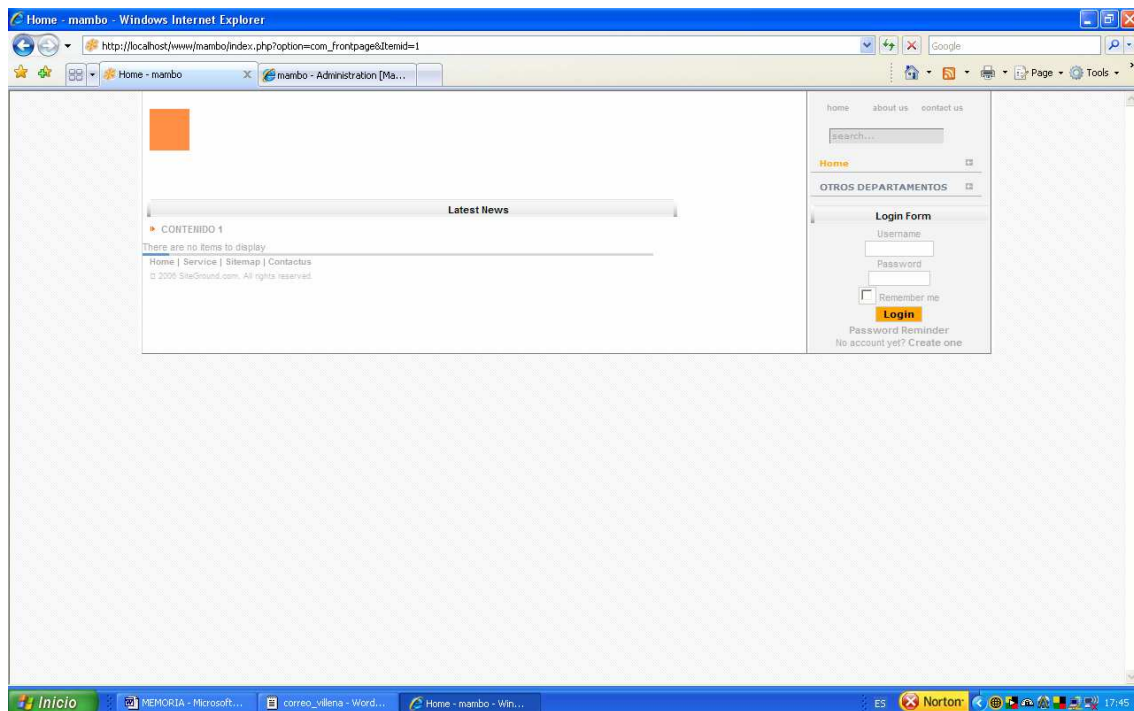


Figura 26: Vista de modificaciones del Template.

6. Ahora hay que ir añadiendo el resto de secciones, categorías y contenidos según la estructura que se desea tener. Si está todo añadido pero nos damos cuenta que en el menú principal no aparecen las secciones en el orden que deberían, por ejemplo justo detrás de “OTROS DEPARTAMENTOS” está “BBDD Conocimiento” cuando esta última debería estar al final del todo se puede acceder al **Menu> Main Menu** y con las flechas de la columna “Reorder” colocar las secciones de la manera deseada. Una vez completada la estructura con todas las secciones, categorías y contenidos que faltaban, se obtiene una página principal como la de la figura.

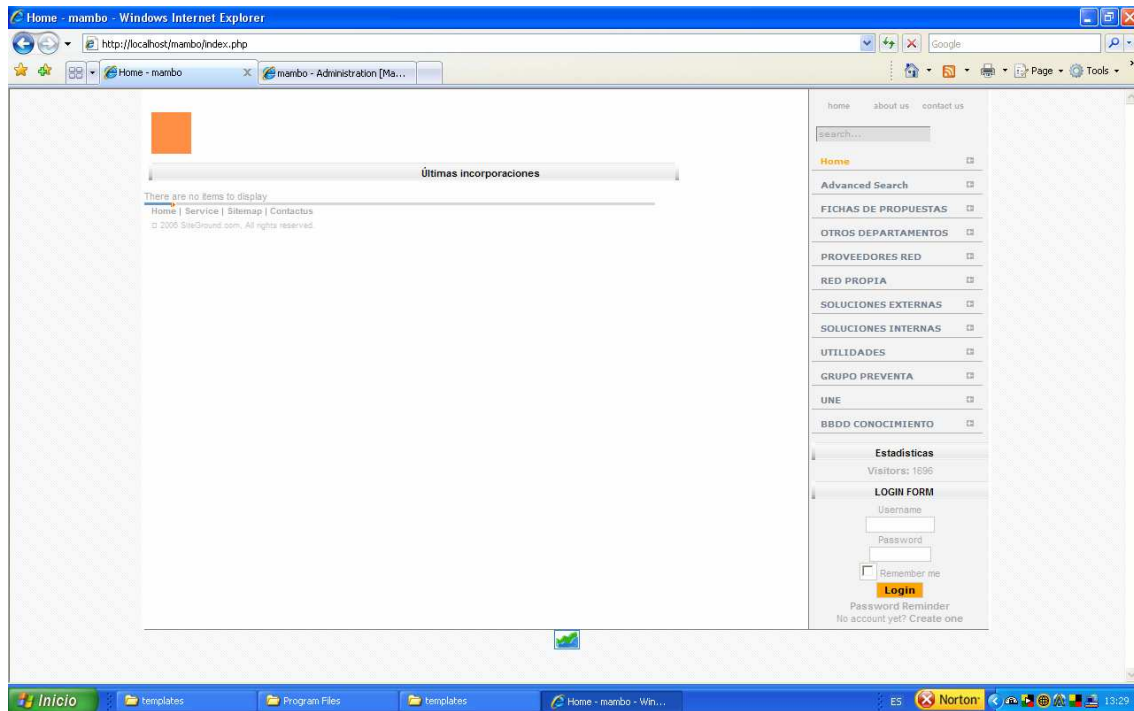


Figura 27: Vista del Template con estructura creada.

4.4 Alta de usuarios.

Esta página que se vería es a la que se accede por defecto sin haber accedido. Es decir, un usuario sin acceder no puede acceder a la información, ni tan siquiera a las últimas noticias. Para que los usuarios puedan acceder y acceder a la información tienen que ser dados de alta en el registro de usuarios a través de la interfaz de administrador. Para ello se pulsa en el icono de “Users” que aparece en la parte principal de la interfaz de administrador, se pulsa “New” y ahí se completan los campos necesarios. En “Group” se tiene que marcar a que grupo pertenece el usuario, ya que en función del perfil que se le dé a cada usuario podrá realizar unas u otras tareas. Como ya se decidió previamente, todos los ingenieros serán editores, de tal forma que puedan acceder a toda la información, hacer ciertas modificaciones sobre contenidos ya publicados y publicar nuevos contenidos bajo supervisión del administrador. Una vez que están todos los usuarios dados de alta al acceder al control de usuarios se debería ver una vista como la de la figura.

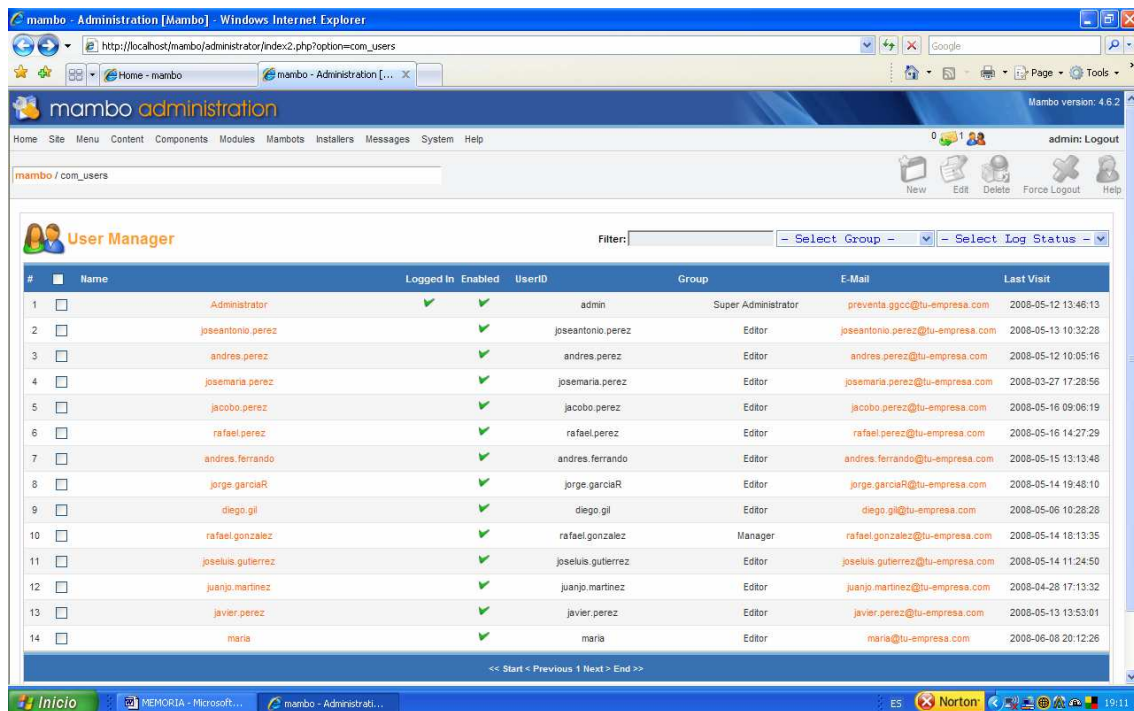


Figura 28: Vista de usuarios.

En este punto los usuarios ya pueden acceder a distintos tipos de información ubicada en la Intranet incluyendo enlaces a páginas Web o a documentos. Una vista de la Intranet buscando información general podría ser como la de la figura.

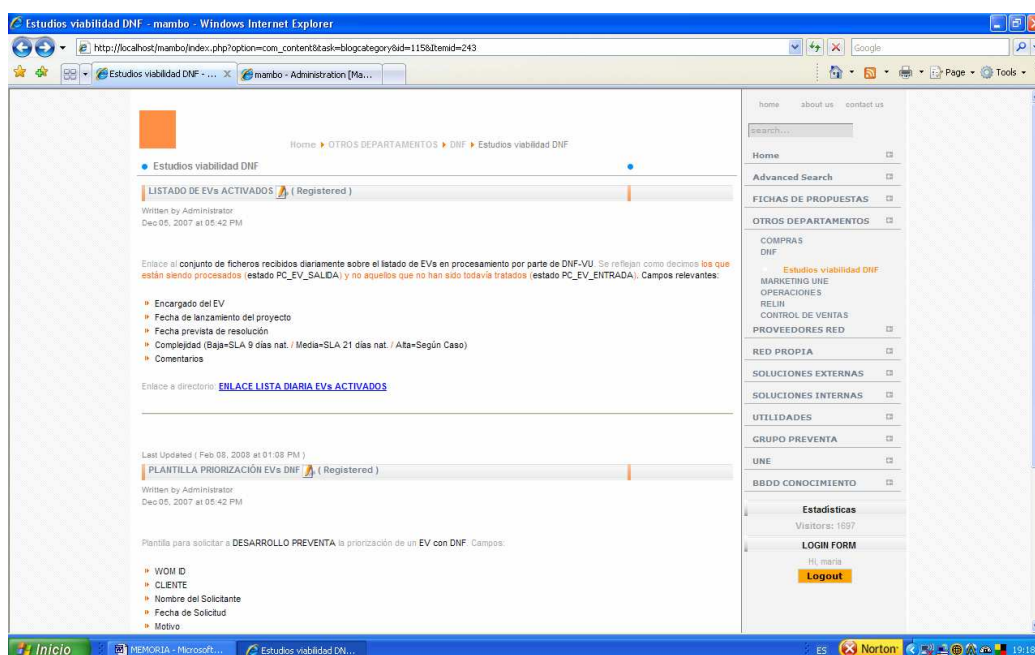


Figura 29: Vista del Template por un usuario tipo editor.

4.5 Creación de base de datos y aplicación de Soluciones Externas

Ahora falta la parte de desarrollo en PHP [3], [4] y [7]. Esta parte está basada en una base de datos de soluciones y partners. Es decir, con la información que se tenía de las distintas soluciones que se tenían la información necesaria para poder crear en la base de datos de Mambo nuevas tablas con información relativa a estas soluciones. Para ello se crearon tres tablas.

- Soluciones: almacena información sobre el tipo de solución. Su clave primaria es el ID pero también está relacionada con las tablas Partners y Sets (ver figura).

Sus campos son:

- ID: Es el identificador de cada solución. Es de tipo entero. Es la clave de esta tabla.
- Solucion: Corresponde al ID de la tabla “SETS”. Es de tipo entero.
- Integrador: Corresponde al ID de la tabla “PARTNERS”. Es de tipo entero.
- Escenario_Aplicacion: Explica en qué circunstancias es aplicable esta solución. Es de tipo cadena de caracteres.
- Productos_Relacionados: Los productos que pueden tener algún tipo de relación con dicha solución. Es de tipo cadena de caracteres.
- Ejemplo_Precios: Algunos precios que ya han sido utilizados para este tipo de solución anteriormente. Es de tipo cadena de caracteres.
- Documento_Tecnico_Tipo: Algún tipo de documento técnico (RFC, manual...) que esta solución tiene asociado. Es de tipo cadena de caracteres.
- Oferta_Tecnica_Tipo: Alguna oferta técnica que ha sido realizada utilizando esta misma solución. Es de tipo cadena de caracteres.
- Configurator_Precios: Cálculos que se tienen que pueden dar una idea del precio de la solución completa. Es de tipo cadena de caracteres.
- Palabras_clave: Lista de palabras relacionadas con esta solución, que se almacenan en este campo pero que no se muestran al usuario. Este campo existe porque es el campo que se utiliza para hacer la búsqueda de las soluciones. Es de tipo cadena de caracteres.
- Tipo_Compras: Es de tipo enumerado (sólo puede ser ‘A’, ‘B’ o ‘C’).

- Fecha: Fecha en que se ha introducido esta solución en la base de datos. Es de tipo cadena de caracteres.
- Partners: Almacena información relativa a un proveedor. Su clave primaria es el ID. Sus campos son:
 - ID: Es el identificador de cada proveedor/Partner. Es la clave de esta tabla. Es de tipo entero. Tiene el mismo valor que el campo “Integrador” de la tabla “SOLUCIONES”.
 - Integrador: Es el nombre del proveedor. Es de tipo cadena de caracteres.
 - Contacto_Comercial: Nombre o nombres de las personas encargadas de la parte comercial de dicho proveedor. Es de tipo cadena de caracteres.
 - Telefono_Comercial: El teléfono/s de la persona/s del camp anterior. Es de tipo numérico.
 - Mail_Comercial: El correo electrónico de la persona encargada de la parte comercial de este proveedor. Es de tipo cadena de caracteres.
 - Contacto_Tecnico: Nombre o nombres de las personas encargadas de la parte técnica de dicho proveedor. Es de tipo cadena de caracteres.
 - Telefono_Tecnico: El teléfono/s de la persona/s del camp anterior. Es de tipo numérico.
 - Mail_Tecnico: El correo electrónico de la persona encargada de la parte técnica de este proveedor. Es de tipo cadena de caracteres.
 - Productos_relacionados_1: Esta tabla contiene 6 campos análogos a este (el último se llama “Productos_relacionados_6”). Cada uno de estos 6 campos almacena un producto que ofrece este proveedor. Es de tipo cadena de caracteres.
 - URL: Página Web del proveedor. Es de tipo cadena de caracteres.
 - logo: Enlace Web que apunta al logotipo de este proveedor. Es de tipo cadena de caracteres.
 - Comentarios: Cualquier tipo de comentario, idea o experiencia previa que se haya tenido con este proveedor. Es de tipo cadena de caracteres.
 - Fecha: Fecha de inserción en la base de datos de este proveedor. Es de tipo cadena de caracteres.

- Sets: Almacena únicamente el tipo de solución que es (CISCO, PABX, HOUSING-HOSTING...). Sus campos son:
 - ID: Es el identificador de cada tipo de posible solución que es. Es la clave de esta tabla. Tiene el mismo valor que el campo Solución de la tabla “SOLUCIONES”.
 - Nombre: Tipo de solución que es. Es de tipo cadena de caracteres.

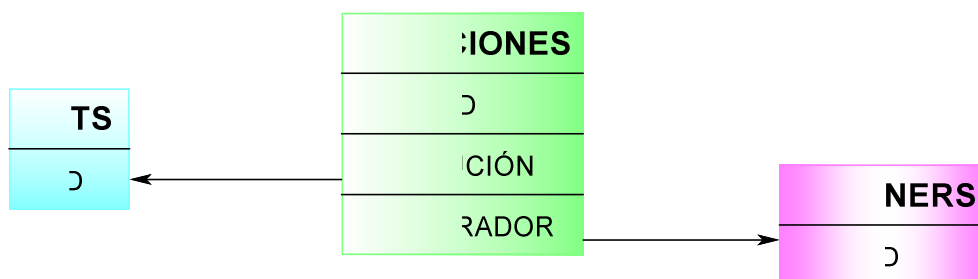


Figura 30: Esquema de tablas de Soluciones Externas con sus claves.

Todas estas tablas se crean en la misma base de datos de Mambo de tal manera que cuando se hace un backup de Mambo (imágenes, módulos, componentes de Mambo) también se esta haciendo un backup de estas tablas. De hecho al acceder al “phpMyAdmin” a través de WAMP se pueden distinguir claramente las tablas nuevas que se han creado de las que venían con Mambo, ya que todas las tablas que vienen por defecto en Mambo llevan el prefijo “mos_”.

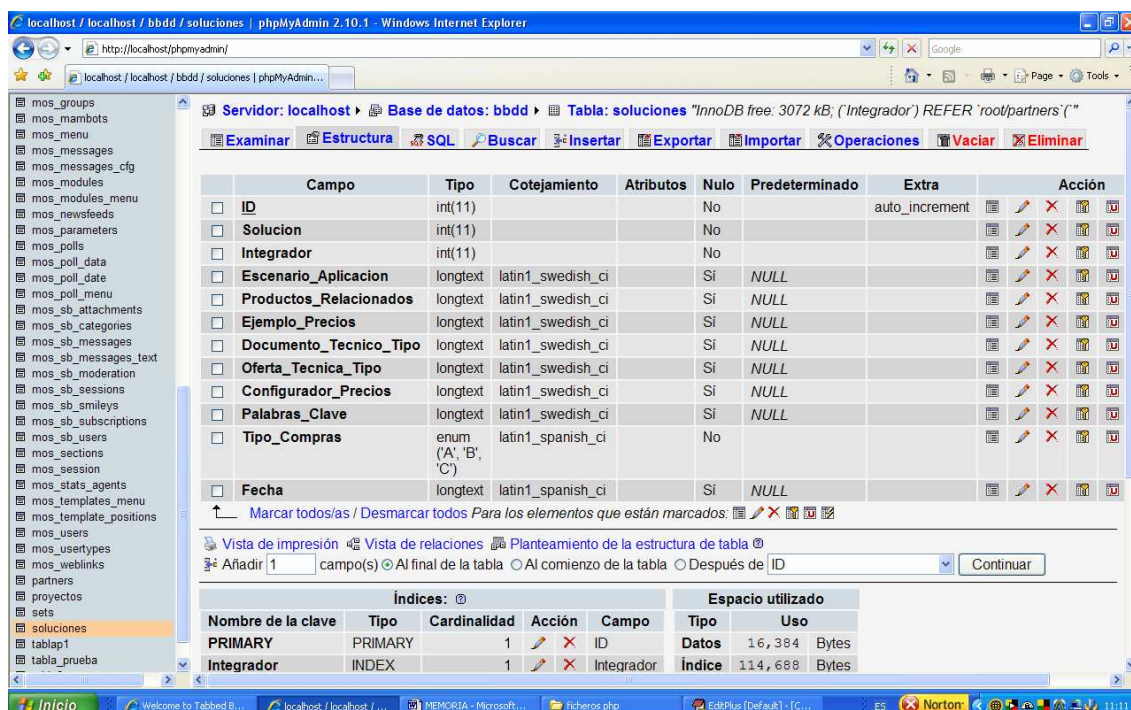
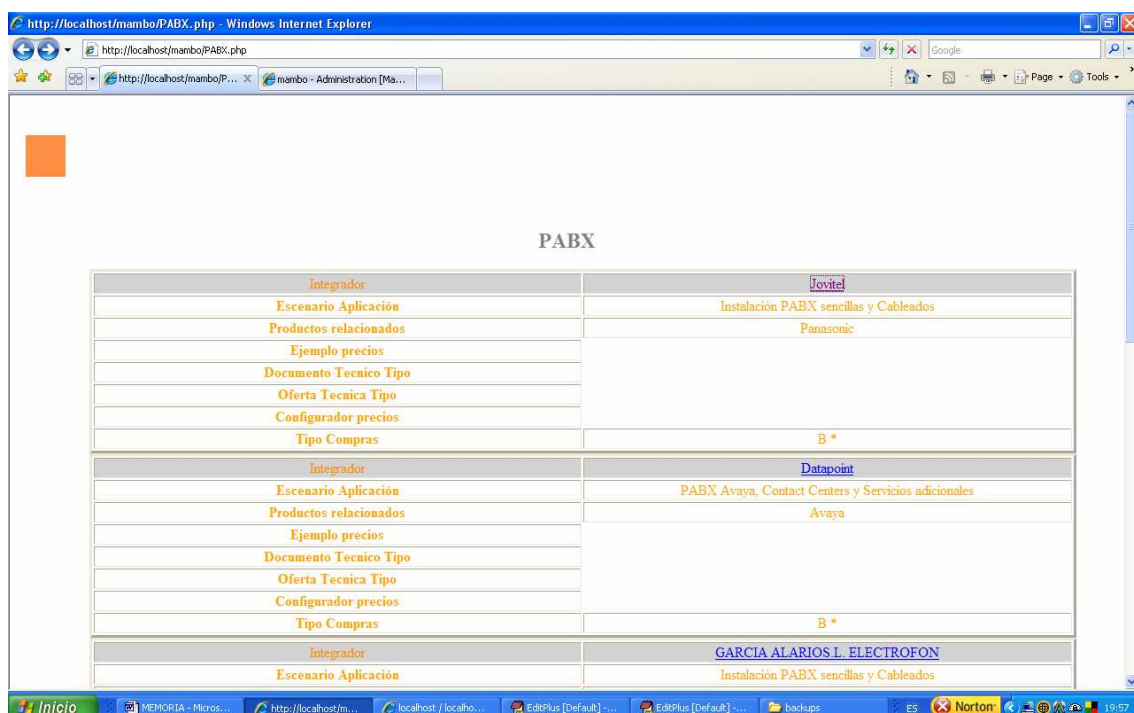


Figura 31: Vista del interfaz de la base de datos, tabla Soluciones.

El funcionamiento que se quiere lograr a través de estas tablas es el siguiente. Si por ejemplo se quiere información relativa a centralitas PABX, se accede a **Soluciones Externas>PABX** y ahí tiene que haber un contenido específico dónde pinchando en un enlace aparecerá una nueva ventana con las soluciones relativas a PABX. En caso de que no hubiese soluciones relativas a dicha tecnología debería aparecer un mensaje informativo al respecto. En el caso de que haya información relativa a dicha solución aparecerá una página como la siguiente.



PABX	
Integrador	Jovitel
Escenario Aplicación	Instalación PABX sencillas y Cableados
Productos relacionados	Panasonic
Ejemplo precios	
Documento Técnico Tipo	
Oferta Técnica Tipo	
Configurador precios	
Tipo Compras	B *
Integrador	Datapoint
Escenario Aplicación	PABX Avaya, Contact Centers y Servicios adicionales
Productos relacionados	Avaya
Ejemplo precios	
Documento Técnico Tipo	
Oferta Técnica Tipo	
Configurador precios	
Tipo Compras	B *
Integrador	GARCIA ALARIOS L. ELECTROFON
Escenario Aplicación	Instalación PABX sencillas y Cableados

Figura 32: Lista de soluciones tipo PABX.

En caso contrario, algo como la siguiente figura.



Figura 33: Vista en el caso de que no existan soluciones PABX.

En el caso de que haya soluciones, el nombre del Integrador es a su vez un enlace a otra página con información relativa a él.

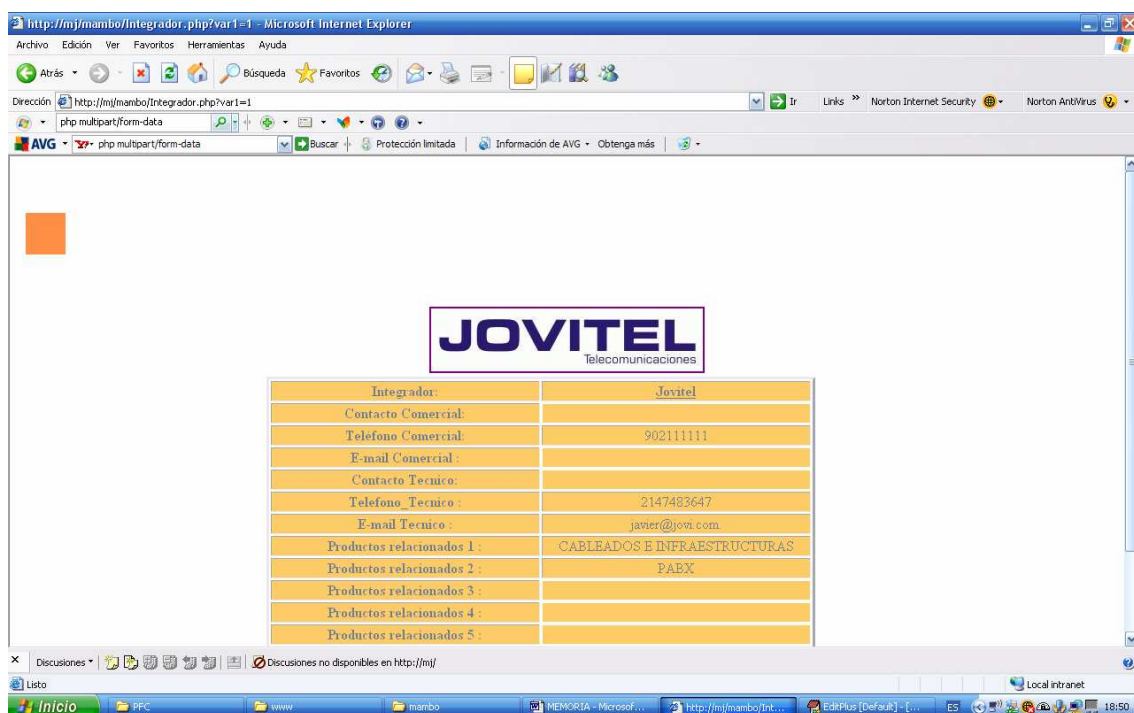


Figura 34: Vista de un integrador concreto.

Y aquí a su vez si se pulsa encima del nombre del integrador o del logotipo de la empresa, se redirecciona hacia la página Web oficial.

Todas estas páginas programadas en PHP llevan su correspondiente archivo CSS para que la imagen sea también corporativa. También se añaden flechas para volver hacia atrás para comodidad del usuario.

Por otro lado en la jerarquía de ficheros, todos estos ficheros .php van justo por debajo de “./mambo/”. En esta parte hay un PHP para cada solución. Una vez que se escoge una solución, hay un PHP único para manejar todos los integradores se llama **Integrador.php**.

Está dividida en dos partes:

- i. Soluciones Externas: en esta categoría aparece una lista con todas las posibles soluciones externas disponibles. En cada solución aparecerá un contenido relativo a esta solución de cualquier tipo (.doc, .xls...). Se desea que además por cada solución aparezca un listado de todos los fabricantes que proveen de esa solución externa a la compañía.
- ii. BBDD Conocimiento: se nutre de la misma base de datos de la anterior más otras tablas y permite que la búsqueda o la inserción la haga directamente el usuario.

4.6 Base de datos de conocimiento.

La parte de la base de datos de conocimiento (BBDD Conocimiento) está dividida a su vez en tres partes: Soluciones Externas, Proyectos y EVs. A la vez cada una de ellas esta subdivida en otras dos categorías: buscar e insertar.

Cada una de estas categorías tiene su propio fichero PHP. Es decir:

- Soluciones Externas
 - Buscar→**buscarS.php**

- Insertar→**InsertarS.php**
- Proyectos
 - Buscar→**buscarP.php**
 - Insertar→**Insertar.php**
- EVs
 - Buscar→**buscarE.php**
 - Insertar→**InsertarE.php**

El PHP puede aparecer directamente incrustado en la página o habrá que pulsar un enlace para acceder a el. En los casos en que haya que pinchar un enlace para acceder al PHP no se necesita ningún tipo de configuración especial. Es decir, se crea el PHP se sitúa dentro de la carpeta “./mambo/” y en el contenido que se quiere que aparezca se escribe el enlace.

El proceso es diferente cuando se quiere que el código aparezca incrustado en el contenido. Para ello habrá que usar un “mambot” específico de Mambo denominado “mosaddphp”. Para que este “mambot” se ejecute es necesario colocar el fichero **mosaddphp.php** en “./mambo/mambots”. Este código es el siguiente:

```

<?php
// $Id: mosaddphp.php,v 1.01 2003/12/03 13:30:00 Neni Exp $
/**
 * mosaddphp mambot
 * @by Neni (neni@synvox.ch)
 * @package Mambo Open Source
 * @Copyright (C) 2000 - 2003 Miro International Pty Ltd
 * @ All rights reserved
 * @ Mambo Open Source is Free Software
 * @ Released under GNU/GPL License : http://www.gnu.org/copyleft/gpl.html
 * @version $Revision: 1.01 $
 * @added output buffering (many thanks to chipjack for suggestion)
 **/

defined( '_VALID_MOS' ) or die( 'Direct Access to this location is not allowed.' );

// $row is made available by the calling function

$mosaddphp_entrytext = $row->text;
$mosaddphp_matches = array();
if (preg_match_all("/{mosaddphp:.+?}/", $mosaddphp_entrytext, $mosaddphp_matches,
PREG_PATTERN_ORDER) > 0) {
    foreach ($mosaddphp_matches[0] as $mosaddphp_match) {
        $mosaddphp_output = "";
        $mosaddphp_match = str_replace("{mosaddphp:", "", $mosaddphp_match);
        $mosaddphp_match = str_replace("}", "", $mosaddphp_match);
        $mosaddphp_params = array();
        $mosaddphp_params = explode(",", $mosaddphp_match);
        if (file_exists("mosaddphp/" . $mosaddphp_params[0])) {
            $mosaddphp_file = "mosaddphp/" .
array_shift($mosaddphp_params);

```

```

        ob_start();
        include($mosaddphp_file);
        $mosaddphp_output .= ob_get_contents();
        ob_end_clean();
    }
    $mosaddphp_entrytext = preg_replace("/{mosaddphp:.+?}/",
$mosaddphp_output, $mosaddphp_entrytext, 1);
    }
    $row->text = $mosaddphp_entrytext;
}
?>

```

1. Ahora quedaría crear una carpeta denominada también “*mosaddphp*” que se colocará justo por debajo de “*./mambo/*” y que contendrá todos los ficheros PHP que se quiere que aparezcan como contenido normal de Mambo. En el propio código de estos ficheros no hay que indicar nada especial que indique este hecho. El código no influye en eso. Es decir, para que aparezcan como contenido de mambo solo es necesario que el “mambot” **mosadphp** esté correctamente ubicado y que la carpeta de “*./mosaddphp/*” que contiene los ficheros incrustados en los contenidos de Mambo también esté correctamente ubicada debajo de “*./mambo/*”.

Para crear un contenido en el que lo que aparezca en realidad sea código PHP se debe hacer lo siguiente. Primero serán los pasos habituales para crear cualquier contenido, seleccionando la sección y categoría donde se quiere que aparezca y el nombre que se le dará. Entonces hay que colocarse donde se escribe el contenido y escribir:

{mosaddphp: <fichero>.php}

Donde <fichero>.php es el nombre del fichero PHP que se quiere que se ejecute al acceder a ese contenido y que debe estar colocado en “*./mambo/mosaddphp*”. Al introducir el contenido como administrador se muestra la figura siguiente.

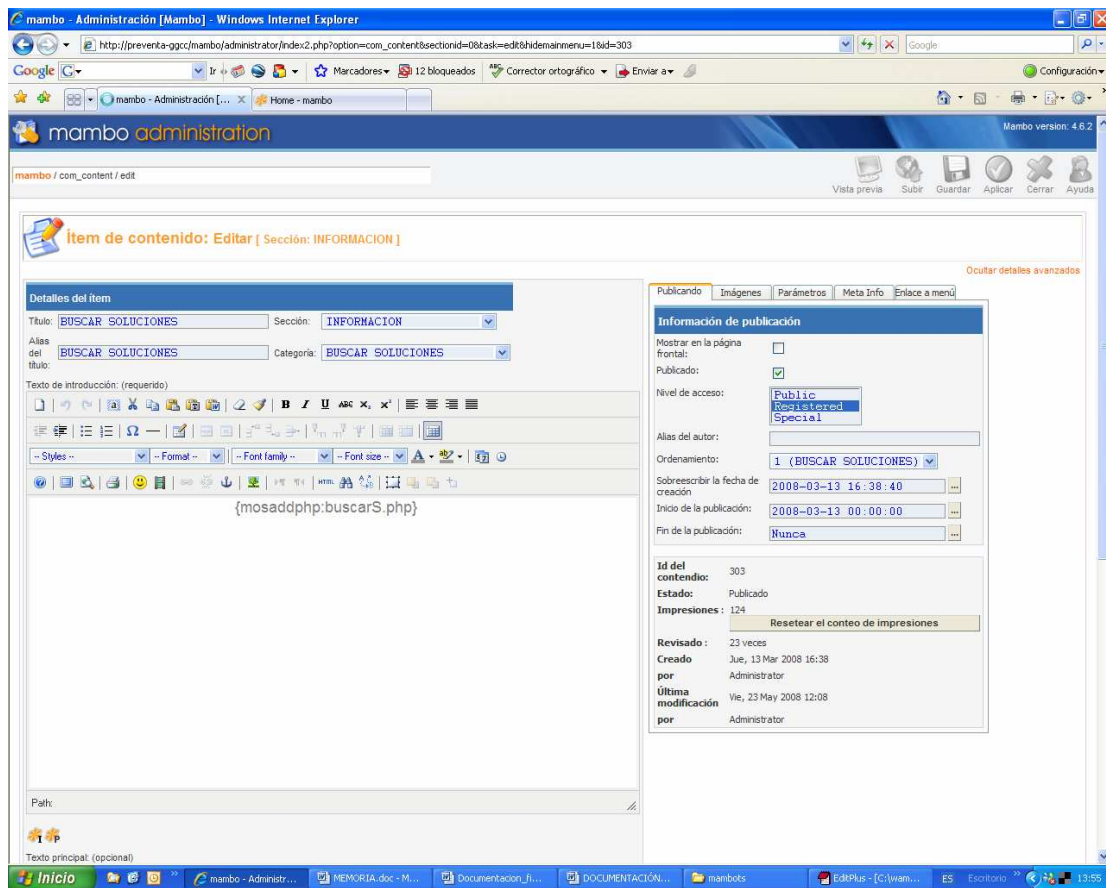


Figura 35: Imagen de cómo incrustar el código PHP.

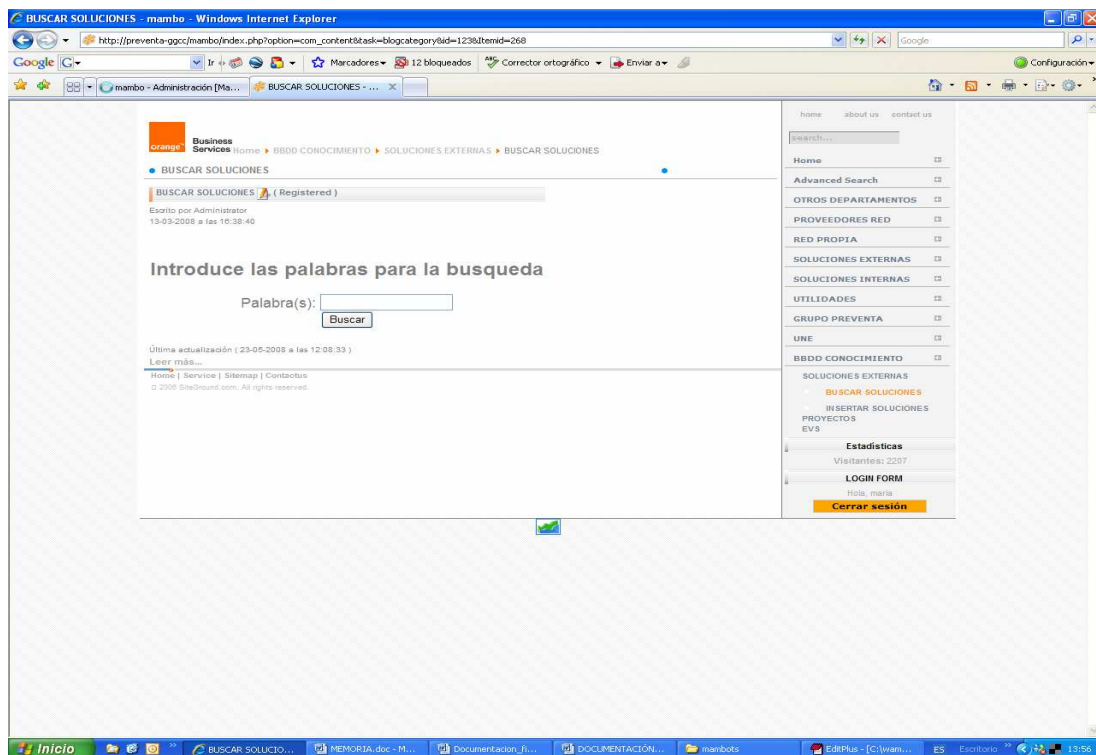


Figura 36: Código PHP incrustado en el Template.

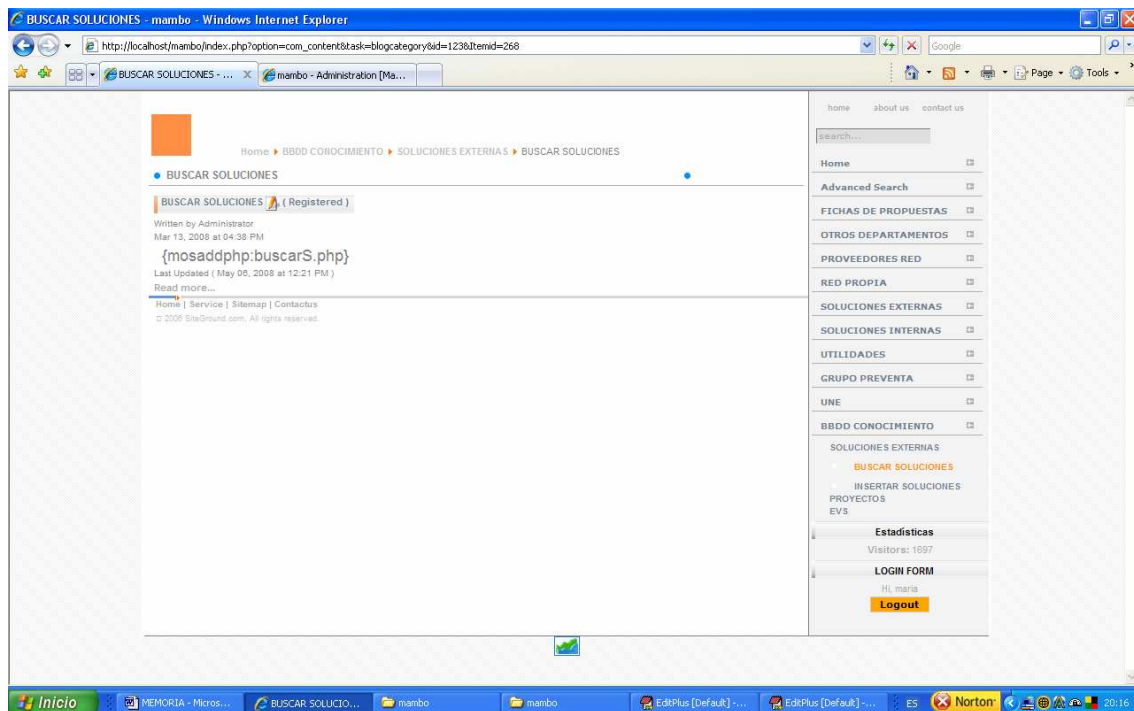


Figura 37: Contenido sin que este correctamente hecho el mosaddphp.

La categoría **BBDD CONOCIMIENTO > SOLUCIONES EXTERNAS** contiene la misma información que la sección con este nombre y se utiliza las mismas tablas de la misma base de datos. La diferencia entre ambas radica en que mientras la sección Soluciones Externas anteriormente comentada, solo permite hacer búsqueda por las tecnologías que aparecen en cada una de las categorías de dicha sección. Por otra parte la categoría **BBDD Conocimiento > Soluciones Externas**, permite por un lado buscar cualquier solución y además permite insertar nuevas soluciones. Para poder insertar nuevas soluciones, habrá unos campos que sean obligatorios y otros opcionales. También existirá la posibilidad de poder subir ficheros y que de forma automática al almacenar la información el fichero quede situado en un lugar que todo el mundo lo pueda descargar, para lo cual será necesario que se cree de manera automática el enlace que quede almacenado en la base de datos que permita realizar posteriormente la descarga.

La categoría de Proyectos almacena información sobre los proyectos que ya se han realizado. De esta manera se puede ver fácilmente qué proyectos se han hecho en qué empresas y qué ingenieros los han llevado a cabo. Aparte de permitir buscar por el nombre

del proyecto, también permite insertar nuevos proyectos. Para esta categoría ha sido necesario crear las tablas de:

- **Proyectos:** cada entrada almacena información relativa a un proyecto en concreto. Por cada entrada tiene un identificador único y campos como nombre del cliente, fecha de entrega, si se ganó o no, ingenieros que intervinieron...Sus campos son:
 - ID: Identificador del proyecto. Es de tipo numérico. Clave de esta tabla.
 - Nombre: Nombre del proyecto. Es de tipo cadena de caracteres.
 - Cliente: Nombre del cliente para el que se realizó el proyecto. Es de tipo cadena de caracteres.
 - Fecha_Entrega: Fecha en la que se entregó el proyecto al cliente. Es de tipo cadena de caracteres.
 - Ingenieros_relacionados: Nombres de los ingenieros que trabajaron en este proyecto. Es de tipo cadena de caracteres.
 - Breve_descripción: Explicación corta de en qué consiste el proyecto. Es de tipo cadena de caracteres.
 - Ganado_NoGanado: Valor que indica si el proyecto se ganó o no. Es de tipo enumerado, es decir, solo puede tener dos valores: 'Ganado'/'NoGanado'.
 - Enlace_Documentacion: Enlace a la carpeta de red dónde se encuentra la documentación del proyecto. Es de tipo cadena de caracteres.
 - Caracteristicas_especiales: Explicación de algunas características concretas o especiales de este proyecto y que pueden ser de interés para otros proyectos. Es de tipo cadena de caracteres.
 - Palabras_Clave: Palabras importantes relacionadas con este proyecto. No se muestran al usuario, es el campo con el que se hace la comparación a la hora de buscar. Es de tipo cadena de caracteres.
 - Fecha: Fecha en la que se almacenó el proyecto en la base de datos. Es de tipo cadena de caracteres.
- **Ingenieros:** cada entrada contiene un identificador único y un nombre y apellidos de un ingeniero. Esta tabla también es referenciada por la tabla de EVS. En cada entrada de la tabla EVS también hay un campo que es nombre de ingeniero. Sus campos son:
 - ID: Identificador del ingeniero. Es de tipo numérico.

- Nombre: Nombre del ingeniero que almacenó el proyecto en la base de datos. Es de tipo cadena de caracteres.

La categoría de EVS, de manera análoga a las anteriores, almacena información respecto a estudios de viabilidad. Incluye información tal como quién ha solicitado el estudio, cuánto ha tardado en ser respondido, para qué empresa era, si fue concedido o no... Para esta categoría ha sido necesario crear las tablas de:

- EVS: cada entrada contiene información relativa a un estudio de viabilidad realizado.

Los campos de esta tabla son:

- ID: Identificador del estudio de viabilidad. Es la clave de esta tabla. Es de tipo numérico.
- Nombre_Cliente: Nombre del cliente para el que se realizó este EV. Es de tipo cadena de caracteres.
- Nombre_Ingeniero: Nombre del ingeniero que realizó el estudio de viabilidad. Es de tipo cadena de caracteres.
- Breve_Descripcion: Corta explicación sobre el EV. Es de tipo cadena de caracteres.
- Palabras_Clave: Palabras importantes relacionadas con este proyecto. Este campo no se muestra al usuario, es para realizar la comparación en el momento de la búsqueda. Es de tipo cadena de caracteres.
- Num_EV: Número de EV. Los EVs tienen una numeración propia independientemente del ID. El ID es para esta tabla, pero el número de EV es un número por el que la empresa localiza el EV. Es de tipo cadena de caracteres.
- Enlace_Documentacion: Enlace a la carpeta de red dónde se encuentra la documentación del EV. Es de tipo cadena de caracteres.
- Fecha: Fecha en la que se almacenó el EV en la base de datos. Es de tipo cadena de caracteres.

Tanto en las categorías de Soluciones Externas como de Proyectos era necesaria la opción de que al insertar se pudiesen subir ficheros. Así, al buscar ese proyecto o solución externa, se pudiese ver o descargar ese fichero. Para ello en ambos casos se utilizará el mismo procedimiento. Para empezar en el formulario a través del cual se permite subir

fichero (**InsertarS.php** para Soluciones Externas e **Insertar.php** para proyectos) el formulario debe contener en la cabecera:

enctype="multipart/form-data"

Este atributo es necesario en el formulario para poder subir en un mismo formulario datos y archivos. También es importante tener presente, que hay un atributo input para ficheros específico, es el siguiente.

<input type="file" name='fichero1'>

El uso de estos atributos en el formulario, permiten que se pueda subir desde un mismo formulario tanto datos como archivos. En los campos donde se permite subir archivos aparece un botón “Examinar...”, que una vez que es pulsado permite recorrer carpetas para seleccionar el fichero que se desea subir a la red.

Con todos los datos completados por el usuario en el formulario, hay que almacenar estos datos. Una vez hecho esto, hay que hacerlos visibles. Es decir, una vez que un usuario ha almacenado, por ejemplo un proyecto, inmediatamente después el usuario tendría que poder buscar a través del buscador de proyectos esta información y tendría que estar accesible, tanto para él como para el resto de usuarios. Hay que pensar que cuándo un usuario busca una información que tiene asociada un fichero, lo que el usuario debe de ver de ese fichero, es el enlace para descargarlo. Por lo tanto, los formularios que tienen la opción de subir ficheros tienen que poder hacer tres funciones:

1. Almacenar los datos del formulario de la manera habitual.
2. Subir el fichero a una carpeta en red a la que tengan acceso todos los usuarios.
3. En el campo de la base de datos destinado al fichero, se tiene que almacenar el enlace a este fichero. Por lo tanto, en la base de datos se almacena una cadena de caracteres que representa la ruta hasta el propio fichero y que permite que el usuario se lo pueda descargar.

La parte de almacenar los datos del formulario en la base de datos es inmediata. Para poder subir el fichero a una carpeta en red a la que puedan acceder todos los usuarios, primero habrá que crear esa carpeta. Esta carpeta se crea con este uso exclusivo, almacenar los archivos subidos por los usuarios.

En este caso, la carpeta se llamará “*ficheros*” y estará ubicada dentro de la carpeta “*./mambo*”. La ruta completa a la carpeta en este caso sería:

“C:\wamp\www\mambo\ficheros”

Para poder subir un fichero a la red, se hace a través de la sentencia que aparece a continuación.

```
$fichero1=$_FILES['fichero1']['name'];
```

En los formularios dónde se permite almacenar ficheros, este campo de “ficheros” es opcional. Por lo tanto en la página programada para el almacenamiento de datos, primero se comprobará si se desea dejar un fichero en la red. Una vez comprobado que si se quiere subir un fichero a la red, el fichero es ubicado en una carpeta temporal. El programa comprueba si ha habido errores en la subida a esta carpeta temporal, y en el caso de que sí haya habido errores no sigue con la subida del fichero. Sino ha habido errores se pasa a colocar el fichero en la carpeta en red. Para ello, primero se forma la cadena completa de caracteres que representa la ruta a esta carpeta en red. A esta cadena se le tiene que añadir al final el nombre del fichero. Con la ruta completa de dónde se va a subir el fichero, se ejecuta la sentencia que finalmente sube el fichero a la red, desde la carpeta temporal dónde se había subido previamente. La sentencia es la siguiente:

```
move_uploaded_file($_FILES['fichero1']['tmp_name'],$nombrefinal).
```

Dónde ‘fichero1’ es el nombre del fichero, ‘\$nombrefinal’ es la cadena completa con la ruta del fichero y ‘tmp_name’ es el nombre de la carpeta temporal dónde estaba el fichero. En caso de que haya algún problema en la subida del fichero, se muestra un mensaje de error por pantalla. Si no ha habido ningún error, en el código se indica poniendo el valor de la variable ‘var1’ a verdadero. El código de todo esto es el siguiente:

```
if($fichero1!=null){
```

```
    $directorio = "\wamp\www\mambo\ficheros/";
```

```
    $temporal = $_FILES['fichero1']['tmp_name'];
```

```
    if(is_uploaded_file($temporal)){
```

```
        $nombrefinal = $directorio.$fichero1;
```

```
        if( move_uploaded_file
```

```
($ _FILES['fichero1']['tmp_name'],$nombrefinal) )
```

```
{
```

```
        echo "El archivo ".$nombre_archivo." se subió correctamente  
a: ".$directorio;  
  
        print "<br> \n";  
        echo "Se ha insertado correctamente";  
        print "<br> \n";  
    }  
    else {  
        echo "Error al copiar archivo.";   
        print "<br> \n";  
    }  
}  
else {  
    echo "Ocurrió algún error al subir el fichero. No pudo  
guardarse.";   
  
    print "<br> \n";  
  
}  
  
$ok1=true;  
}
```

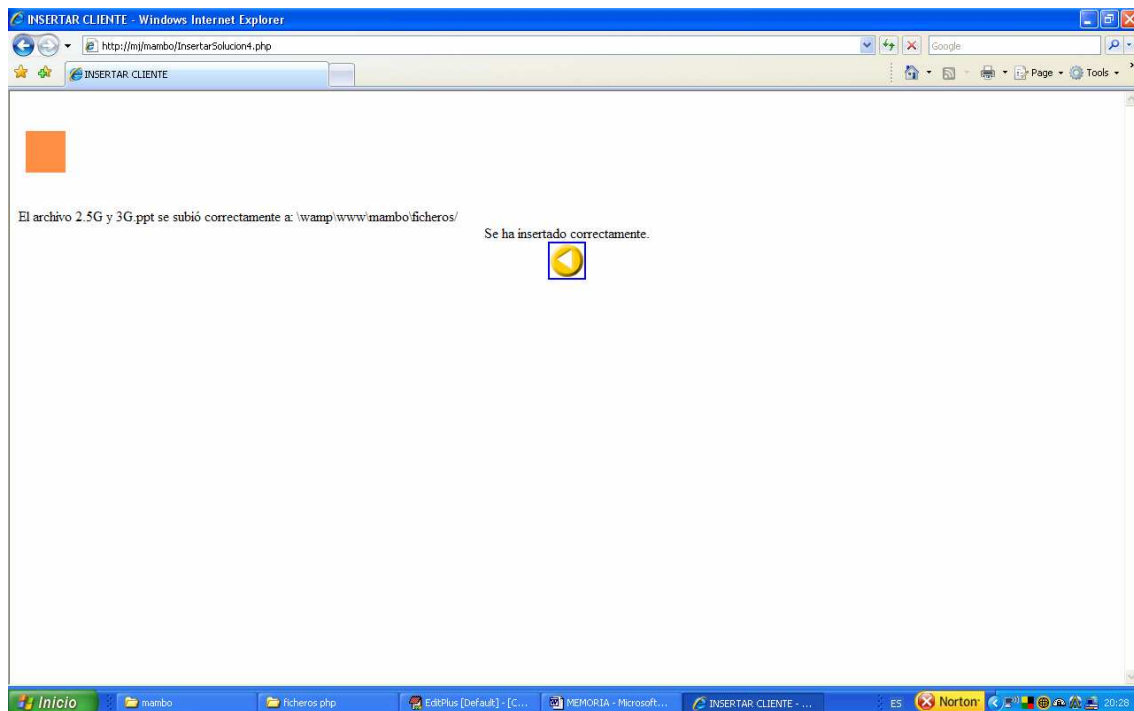


Figura 38: Subida correcta de ficheros e inserción en la base de datos.

Otra manera de comprobar que el fichero se ha subido directamente es yendo a la carpeta “./ficheros” y comprobando que ahora el fichero también está ahí.

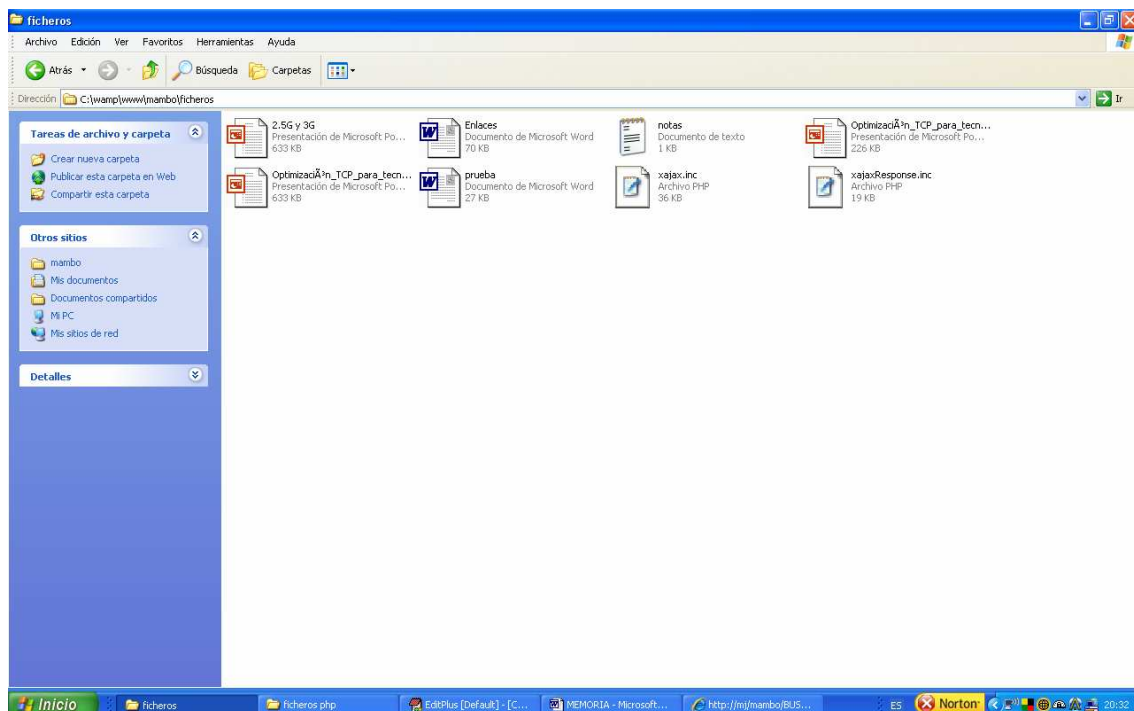


Figura 39: Comprobación de fichero copiado en la carpeta de destino.

La variable \$ok1 indica si el primer fichero de todos los posibles se ha subido correctamente (en el caso de **BBDD> SOLUCIONES EXTERNAS** se permiten subir hasta cuatro archivos desde un mismo formulario, por tanto hay una variable por cada uno de los ficheros que permite subir: \$ok1, \$ok2, \$ok3 y \$ok4). Esta variable(s) se inicializa con valor falso y sólo se cambia este valor cuando el fichero está en la red. La presencia de esta variable(s) está justificada, puesto que una vez que el fichero se ha subido a la red, en la base de datos se almacenan los datos y para almacenar el enlace al fichero se comprueba el valor de dicha variable. Es decir, en el momento de almacenar en la base de datos el enlace al fichero, primero se comprueba que el valor de la variable sea verdadero (el fichero ha sido subido sin problemas) y entonces se almacena el enlace correspondiente a dicho fichero. En caso contrario, en el campo de enlace no se almacena nada. El código de cómo se forma el enlace es el que aparece a continuación.

```
$inicio1="<p>&nbsp;</p><p><a href=".'";
$inicio2="http://mj/mambo/ficheros\\\\";
$final="'. "><u> DESCARGAR!!! </u></a></p>";
```

```
if ($ok1==true){
    $Enlace1=$inicio1.$inicio2.$fichero1.$final;

}
```

Este es un ejemplo con el que se consigue que sea descargable el fichero que previamente se había subido a la red. Se busca en la solución y en la lista de todas las soluciones que hay se busca la que en productos relacionados tiene “OPTIMIZACIÓN TCP” dónde se pulsa “DESCARGAR” y ya se tiene el fichero.

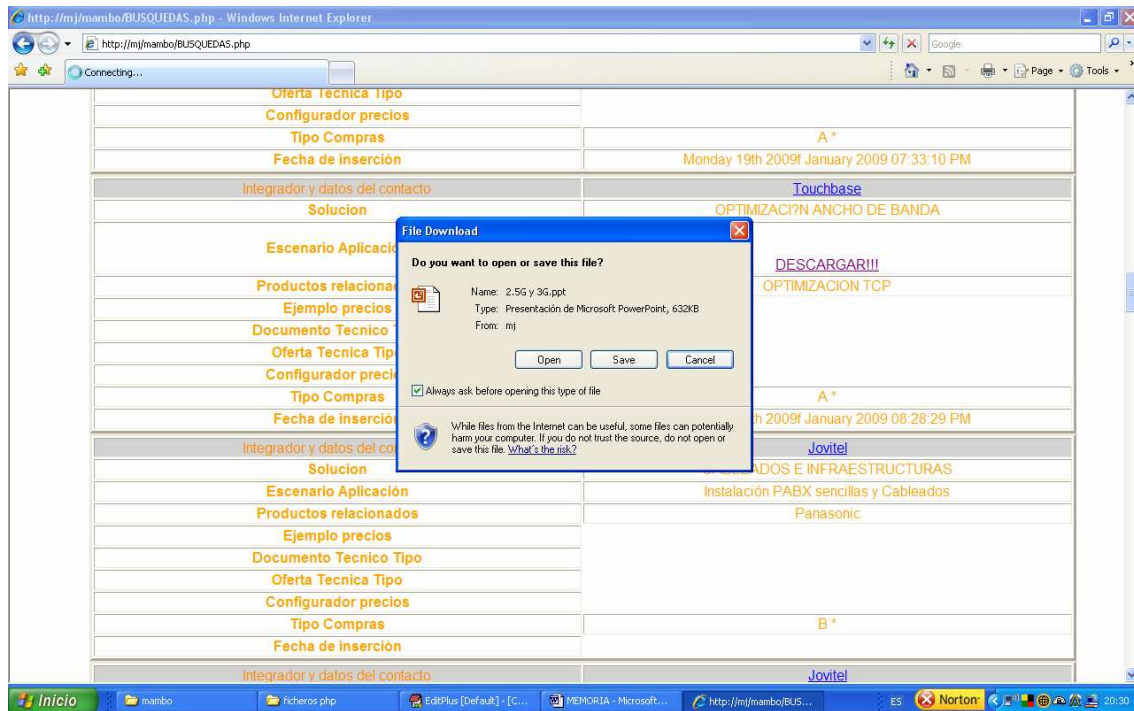


Figura 40: Comprobación de posibilidad de descarga de un fichero subido por los propios usuarios.

4.7 Recarga de desplegables

Para la opción de **Proyectos> Insertar** había un problema. Como en la pantalla que te permite insertar hay varios desplegables que están al final del formulario, cada vez que se seleccionaba un desplegable la página se recargaba por completo, perdiendo así lo que ya estaba escrito en los campos superiores. Se trata de conseguir que cuando se ha escrito algo en los campos superiores y se tiene que seleccionar una parte de un desplegable, los primeros campos sigan guardando esa información y no haya que volver a escribirlos. Con el objetivo de poder recargar los desplegables sin perder la información ya introducida se decidió usar XAJAX, tecnología con la que este problema queda solventado. Para poner en funcionamiento XAJAX hay que realizar las siguientes tareas:

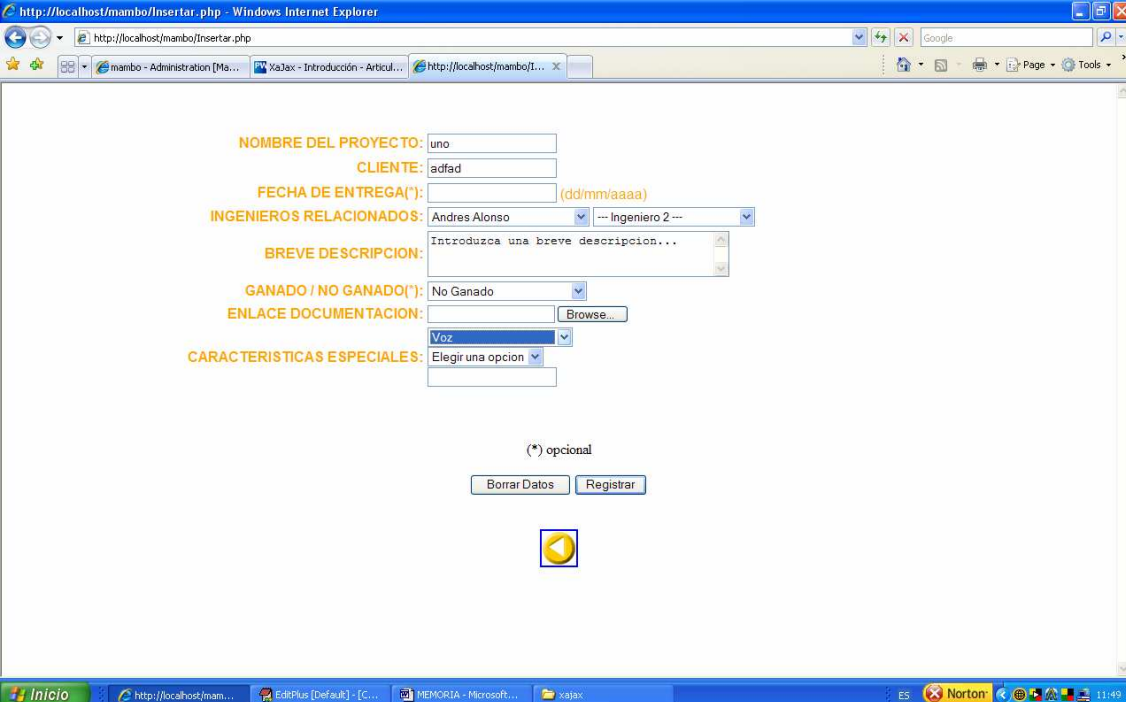
1. En el fichero .php que se va a usar AJAX hay que indicar de alguna manera que se quiere usar. Para ello se escriben las siguientes líneas de código al comienzo del PHP.

```
require ('../xajax/xajax.inc.php');  
$xajax = new xajax();  
$xajax->setCharEncoding('ISO-8859-1');  
$xajax->decodeUTF8InputOn();
```

Con este código se importa la clase XAJAX, se crea un nuevo objeto de esta clase y se define que codificación se quiere que tenga.

Ahora hay que instalar la librería XAJAX correctamente. Para hacerlo correctamente hay que abrir el zip que descargado de la Web de XAJAX y se descomprune manteniendo la estructura de las carpetas. Cuando se tiene la carpeta descomprimida hay que cambiarle el nombre (normalmente indica la versión) por XAJAX. Está carpeta descomprimida XAJAX debe estar dentro de ./WWW. Dentro de esta carpeta hay 3 scripts PHP: xajax.inc.php que contiene la clase xajax, xajaxResponse.inc.php que contiene la clase xajaxResponse y xajaxCompress.php, que reduce de alguna manera el peso de un archivo Javascript.

Para el funcionamiento basta con hacer un require al xajax.inc.php de la carpeta donde se ha instalado que contiene un servidor local o remoto.



The screenshot shows a web browser window with the address bar displaying 'http://localhost/mambo/Insertar.php'. The page contains a form for inserting projects with the following fields and options:

- NOMBRE DEL PROYECTO: Input field with 'uno' entered.
- CLIENTE: Input field with 'adfad' entered.
- FECHA DE ENTREGA(*): Input field with '(dd/mm/aaaa)' as a placeholder.
- INGENIEROS RELACIONADOS: Dropdown menu with 'Andres Alonso' selected and '--- Ingeniero 2 ---' as an option.
- BREVE DESCRIPCION: Text area with placeholder text 'Introduzca una breve descripcion...'.
- GANADO / NO GANADO(*): Dropdown menu with 'No Ganado' selected.
- ENLACE DOCUMENTACION: Input field with a 'Browse...' button.
- CARACTERISTICAS ESPECIALES: Dropdown menu with 'Voz' selected and 'Elegir una opcion' as an option.

Below the form, there is a note '(*) opcional' and two buttons: 'Borrar Datos' and 'Registrar'. A yellow play button icon is located at the bottom center of the form area.

Figura 41: Vista de formulario para insertar proyectos.

5 Implantación del sistema

5.1 Uso del sistema

Esta aplicación está siendo empleada por 15 usuarios de una manera constante con un nivel de aceptación elevado, en un departamento de producción de France Telecom España S.A. Concretamente el departamento en cuestión se encuentra en Unidad Negocio Empresa → Grandes Cuentas → Preventa.

Su utilización ha ido progresando en el tiempo, si bien es verdad que desde el primer momento su acogida fue muy buena. Su uso ha ido aumentando al igual que ha ido aumentando la cantidad de contenidos que podían encontrar en ella.

Al comienzo de su puesta en marcha solo aparecía la estructura de organización de la información (en el menú de la izquierda) y algún contenido en esta estructura. Con el tiempo se fue aumentando el número de contenidos. Además en un principio a todos los usuarios se les asignó el perfil de “Registered” por lo que solamente podían leer la información. Con el tiempo se contempló la posibilidad de que además los usuarios pudiesen opinar o dejar anotaciones sobre la información allí colocada, para compartir esa idea con el resto de compañeros. Por este motivo a todos se les asignó el perfil de editor, pudiendo así comentar los contenidos que el administrador había colgado.

Llegados a este punto, los usuarios hacían un uso diario de la aplicación, aunque este uso estaba limitado exclusivamente a leer contenidos y añadir alguna observación al respecto. Como la veían muy fácil de usar, se propuso la idea de crear un gestor de información dentro de la misma aplicación, para que pudiesen insertar documentaciones completas ellos mismos y al mismo tiempo buscarla.

Esta aplicación en un principio solo se iba a usar para la parte de Soluciones Externas. Una vez implementada esta parte (mediante el uso del lenguaje de programación PHP) se vio la gran facilidad de manejo por parte de los usuarios y cómo mejoraba su eficiencia en el trabajo al poder buscar exactamente lo que necesitaban, por lo que se propuso la idea de extender este formato también a Proyectos y EVs.

En cuanto a la disponibilidad del sistema, es decir la aplicación por sí misma no ha tenido un fallo que la haya dejado inutilizable, pero sí hay que tener en cuenta que al utilizarse como servidor de esta aplicación un ordenador cuyo uso no era exclusivo ha habido ocasiones en que el ordenador ha tenido algún problema y por lo tanto la aplicación ha quedado inutilizada normalmente. En el momento en que el ordenador ha vuelto a estar funcionando correctamente, la Intranet ha vuelto a ser utilizable sin problemas. Además no es necesario arrancar nada al volver a tener disponible el ordenador, ya que WAMP se instala de modo que se inicie en el momento en el que el PC este funcionando correctamente

5.2 Estadísticas

Para poder comprobar quién accedía a la Intranet, y cada cuanto tiempo accedía se usaron dos sistemas de control.

El primero lo proporciona el propio Mambo. A través de la interfaz del administrador se puede acceder a “Users” donde aparece un listado de todos los usuarios registrados con la fecha y hora exacta de la última vez que se han accedido a la Intranet. Esto permite al administrador tener una idea general del uso de la aplicación por parte de los usuarios.

El segundo método utilizado es usar un contador de visitas. Para ello se empleó un servicio modificando el Template de inicio. Añadiendo el código proporcionado por el servicio externo:

```
<!-- Codigo estadisticas-gratis.com ver. 4.4 -->
<table align="center">
  <script language="javascript">
    <!--<link href="/templates/siteground4/css/template_css.css"/>-->
    var id="c1c411935188c5b5ffdf0a46de9583aa";
  </script>
  <script align="center" type="text/JavaScript" language="javascript"
src="http://www.estadisticas-gratis.com/c.js"></script>
  <br>
</table>
<!-- FIN Codigo estadisticas-gratis.com -->
```

El usuario solo veía de esta aplicación un pequeño símbolo en la parte inferior de la página. Al pulsar sobre dicho símbolo se obtiene una información estadística completa de las visitas realizadas a esta página, con un desglose de página visitada por hora, día, semana, mes y año.

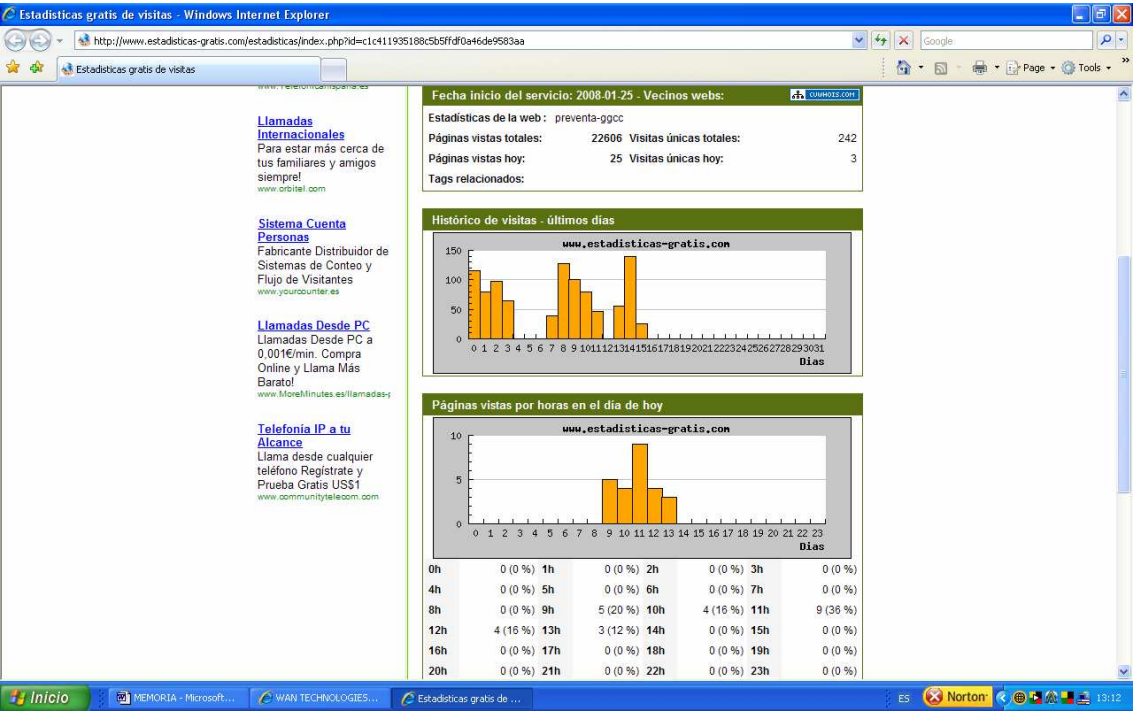


Figura 42: Imagen de estadísticas que se pueden obtener.

Como conclusión se puede destacar que el mayor beneficio aportado sea una mayor implicación de los usuarios en la producción de contenidos. Al ser los propios usuarios los que ven como buscan y usan continuamente dichos contenidos, son ellos los primeros interesados en aportar ideas de cómo debería estar organizada esa información y qué información puede ser más interesante para su trabajo.

6 Conclusiones y trabajos futuros

6.1 Conclusiones

El presente proyecto se ha centrado en el diseño, desarrollo y mantenimiento de la Intranet corporativa, más concretamente en el departamento de Ingeniería Preventa de una empresa durante un año de beca. Para ello se realizó un estudio previo de diferentes elementos a tener en cuenta para la construcción de una Intranet como servidor Web, lenguaje de programación, base de datos, CMS...que en ese momento estaban disponibles. Una vez realizado este estudio previo y teniendo claros los elementos con los que trabajar y objetivos a conseguir, se dedicó gran parte del tiempo del proyecto a la implementación de esta herramienta empresarial. Se ha dedicado especial esfuerzo en conseguir una interfaz que fuese amigable e intuitiva para los usuarios y, a su vez, sencilla y robusta para el administrador. La gente necesitaba una herramienta para compartir información, pero no podía perder demasiado tiempo en aprender, por lo que uno de los requerimientos fundamentales era que fuese intuitiva.

Por lo tanto, el objetivo de este proyecto ha sido cumplir las necesidades específicas y reales de un cliente concreto. Por ello ha primado por encima de otras características la inmediatez de instalación (tanto la primera vez, como en recuperaciones posteriores), la solidez del sistema, la gran capacidad de almacenaje...Por ello, algunas de las conclusiones de este proyecto se derivan del estudio previo y posterior puesta en marcha de un CMS en una situación real. Éstas son:

- La existencia de información y comunidades activas que puedan proporcionar información sobre las tecnologías con las que se trabajan son fundamentales para conseguir trabajar más rápido y con mayor acierto.
- La flexibilidad y extensiones del lenguaje en el que se basa el CMS es indispensable a la hora de ponerse a trabajar en un proyecto donde muchas aplicaciones se conoce que las va a tener que desarrollar personalmente el administrador de la aplicación. Es decir, hoy en día los CMS aportan una gran cantidad de módulos propios, pero es muy posible que sea necesaria alguna función específica y que no haya otra alternativa que tener que desarrollarla. Para esto es fundamental que el lenguaje de programación sea eficaz y sencillo.

6.2 Trabajos futuros

Este Proyecto Fin de Carrera deja abierta la puerta a nuevas aplicaciones que puedan ser útiles para el futuro. Partiendo del trabajo ya realizado se podría pensar en líneas de trabajo como las siguientes:

Debido a que esta aplicación ha sido desarrollada en y para el entorno empresarial, resulta inmediato pensar en poder acceder a dicha aplicación desde dispositivos Blackberry, puesto que los usuarios de la aplicación no se encuentran siempre con el ordenador delante y esta funcionalidad les supondría poder acceder a una gran cantidad de información de forma inmediata desde cualquier lugar donde se encuentren. Habría que ver como interactúa el navegador de la Blackberry con la Intranet y solucionar los posibles problemas que pudieran ocurrir. Un potencial problema es que la Intranet hace uso

Por otro lado se podría pensar en que los propios usuarios pudiesen modificar la información que ya se encuentra disponible en la sección de “BBDD conocimiento”. Es decir, si conocen que hay algún dato nuevo o modificado respecto a alguna de la información de la base de datos, que ellos mismos puedan gestionarlo, sin necesidad de la intervención del gestor de la Intranet. Esto puede ser buena idea para ciertos campos pero no para todos. Por ejemplo, hay informaciones introducidas como descripción de tecnologías y comentarios que escribió una persona con gran conocimiento en el tema por lo que estas secciones de la información no deberían ser modificadas a no ser que sea por esa persona. Sin embargo otros elementos menos técnicos como pudiesen ser datos personales (teléfonos y e-mails de comerciales y técnicos, nuevos logos o URL) sí sería interesante que los usuarios pudiesen cambiarlo. Para ello habría que cosas podrían cambiar los usuarios y qué no. Una vez decidido esto bastaría con crear un nuevo formulario del mismo tipo que el disponible para insertar o buscar pero que en este caso modifique, comprobar que realmente modifica correctamente y publicarlo finalmente en el CMS.

Otra posible mejora sería utilizar como máquina servidora una máquina bastión, es decir una máquina lo más segura posible, por ejemplo como la que se usa como servidor para la Web corporativa de la empresa. Hasta ahora esta Intranet ha estado siempre en una máquina segura, pero al fin y al cabo un ordenador de sobremesa de oficina con todos los riesgos que eso conlleva. Sería aconsejable trasladar la Intranet a otra máquina más segura o esta misma máquina a otro lugar tipo habitación acorazada o algo así donde los niveles de seguridad sean mayores.

Una mejora muy recomendable sería que en la página de inicio de la Intranet apareciese por algún lado unos iconos intuitivos que permitieran cambiar fácilmente de un idioma a otro, como los que hay en muchas páginas Web. Mambo soporta distintos lenguajes por sí mismo, por lo que en principio sería buscar el mambot adecuado y una vez instalado mostrarlo al usuario final. Además, traducir los PHP desarrollados al efecto y la información almacenada. De este modo si en un futuro comenzasen a trabajar empleados que no hablen español no tendrían ningún problema en usar la Intranet como el resto de sus compañeros.

Tal vez sería interesante añadir algún tipo de enlace o RSS para que noticias importantes del mundo de la tecnología apareciesen de forma inmediata en la portada de la Intranet. Para esto habría que pensar bien qué tipo de noticias podrían interesar más y una vez decidido qué tipo de noticias, ver de dónde obtenerlas. Luego solo quedaría ajustar la posición y el tamaño que tendría en la portada. Habría que buscar algún tipo de tamaño fijo o una especie de marco para que este tipo de noticias apareciesen siempre con el mismo formato y ocupando el mismo espacio, sin distorsionar con esto la apariencia del resto de la página.

Programar una funcionalidad de alerta para que en el momento en que se publique un nuevo contenido o se inserte o se modifique una nueva información en la base de datos, los usuarios reciban una notificación en el correo electrónico de forma automática. Ya se ha comprobado que el servidor de correo y la función de enviar correo para una lista de gente funciona. Ahora habría que pensar cómo hacer que la propia Intranet detectase cuándo se ha publicado un nuevo documento. Una vez que se estudie en qué parte del código o de los scripts se da esta detección, habría que programar que justo en dicho momento se envíe automáticamente a la lista de usuarios un correo que será un correo predefinido del tipo “Novedad en la Intranet” de forma que el usuario ni siquiera tenga que navegar en ella para saber si hay alguna novedad.

Otra posible mejora sería la migración de Mambo a Joomla. Aunque realmente Joomla no es más que una extensión de Mambo (los desarrolladores del código de Mambo ahora están en Joomla), sería interesante migrarlo porque parece que Joomla va a publicar en un futuro más extensiones y módulos que Mambo, aunque también parece que existe la posibilidad de que Mambo saque una nueva versión mejorada. Por otra parte, esta migración no parece complicada existiendo en Internet gran información de cómo hacerlo y manteniendo en todo momento la información y estructura que ya se tiene en Mambo.

Otro beneficio sería conseguir hacer copias de seguridad de forma periódica y automática. Aunque el mambot que Mambo tiene para realizar los backup de toda la base de datos es muy eficiente, lo único que le quedaría para ser completamente eficiente es que ni el propio administrador tenga que hacerlo, es decir que a través de un script u otro tipo de aplicación se determinase que de una manera periódica se hiciese un backup de la base de datos, sin necesidad de la colaboración del administrador.

Anexos

Manual de usuario

Para entrar

Acceda a <http://mj/mambo/>, introduzca su login y password. Estos datos para poder logarse en la Intranet son:

- login: toda la cadena que precede el @ del correo electrónico corporativo.
- password: nombre propio de cada usuario. En caso de nombre compuesto solo el primer nombre.

Nota: Si algún usuario quisiera cambiar tanto el login como password, se mandará un correo al administrador de la aplicación para realizar dicho cambio.

Para editar contenidos

La utilidad de ser editores de esta Intranet es que se podrá modificar los contenidos que están colgados, pero sólo el contenido, no la estructura de como están organizados los elementos. Para poder hacer cambios habrá que seguir los siguientes pasos:

1. Seleccione el contenido que quiere modificar o añadir en el apartado correspondiente de la solución o familia de soluciones.
2. Pulse el icono con forma de lápiz que hay en la cabecera del contenido (al pasar el ratón por encima de éste aparecerá la palabra “Editar”).
3. A continuación aparecerá el contenido como en un cuadro de texto arriba y luego abajo otro cuadro de texto. Escriba la información a añadir donde ya está el contenido, es decir, arriba.
4. Una vez añadido todo lo deseado, pulse el botón guardar que está arriba del contenido que tiene forma de “disquete”. Si no se pulsa este botón no se almacenarán los cambios.

Para insertar documentos

Esto se hace insertando un hipervínculo en el contenido.

Los pasos a seguir para insertar un hipervínculo son los siguientes:

1. Escriba el nombre del documento que se quiera colgar en la página correspondiente.
2. Seleccione el nombre y pulse el botón que hay arriba para insertar editar enlaces, similar a dos aros grises.
3. Solo hay que completar en la Sección General el espacio donde dice “Link URL”.
4. Lo más sencillo es dejar el fichero por debajo de “./mambo” . Basta con escribir la ruta desde la carpeta “./mambo” hasta dicho fichero.
5. Para insertar un enlace a Internet basta con escribirlo directamente como:

`http://[url]`

Instrucciones para recuperar la Intranet

1. Instale Wamp.
2. Una vez que Wamp funcione correctamente copie el instalable de mambo en `./wamp/WWW`. Debe descomprimir todo en una sola carpeta.
3. Para saber si wamp está funcionando correctamente, mire el icono inferior derecho, que este todo blanco, semicírculo en blanco.
4. Descomprima el instalable de mambo dentro de `./wamp/WWW/` y cámbiele el nombre, por ejemplo a 'mambo'.
5. Acceda a wamp, localhost y 'mambo' o el nombre que se le haya asignado en el paso anterior ('mambo' en este caso). Al acceder aquí se accede a la instalación del CMS Mambo. Siga los pasos para instalación. Para esto se pueden seguir los pasos del siguiente enlace (aunque en Google aparecen algunas páginas explicando esto):
<http://www.cmsjoomla.com.ar/mambo/instalacion-de-mambo.html>

IMPORTANTE

En la página de instalación que pide el nombre del sitio, realmente se podría poner cualquier nombre, pero será conveniente poner un nombre fácilmente reconocible.

En la penúltima hoja de instalación aparece la URL definitiva por defecto (en caso de haber puesto como nombre mambo):

`http://localhost/mambo/`

es muy importante cambiar esta URL por esta otra:

`http://mj/mambo/`

donde 'mj' es el nombre de la máquina. Se recomienda el nombre de la máquina porque por la IP o localhost hay aplicaciones que no lo reconocen.

Si no se hace este cambio, en esta máquina se verá bien, pero en el resto de las máquinas no se resuelven los enlaces porque todo apunta a localhost.

Por otra parte en esta página también aparece el password de administrador que el propio administrador será quién elija el nombre.

Y la carpeta mossadphp que contiene los ficheros php que aparecerán como código incrustado en

Se coloca en:

./mambo/mambots

Con esto ya estarían disponibles todos aquellos ficheros php cuyo código esta incrustado en los contenidos de la Intranet.

6. Por otro lado hay que volver a colocar los ficheros php que no están dentro de mosaddphp. Estos ficheros son todos aquellos que no están incrustados en el contenido. Estos ficheros simplemente se tienen que colocar en:

./mambo/

7. Para recuperar la parte correspondiente a xajax hay que hacer lo siguiente. Coger la carpeta llamada xajax que está en:

<http://www.desarrolloweb.com/articulos/xajax-libreria-php.html>

y se pega por debajo de mambo, otra vez en:

./mambo/

Ahora la Intranet debería estar recuperada.

Diagrama de clases PHP.

Para aclarar que clases PHP están relacionadas con la Intranet, o más concretamente que clases PHP están involucradas cuando se navega por una u otra parte de ella, los archivos PHP se van a explicar divididos en dos grandes grupos en función de desde que punto de la Intranet son invocados: Soluciones Externas y BBDD Conocimiento.

Soluciones Externas

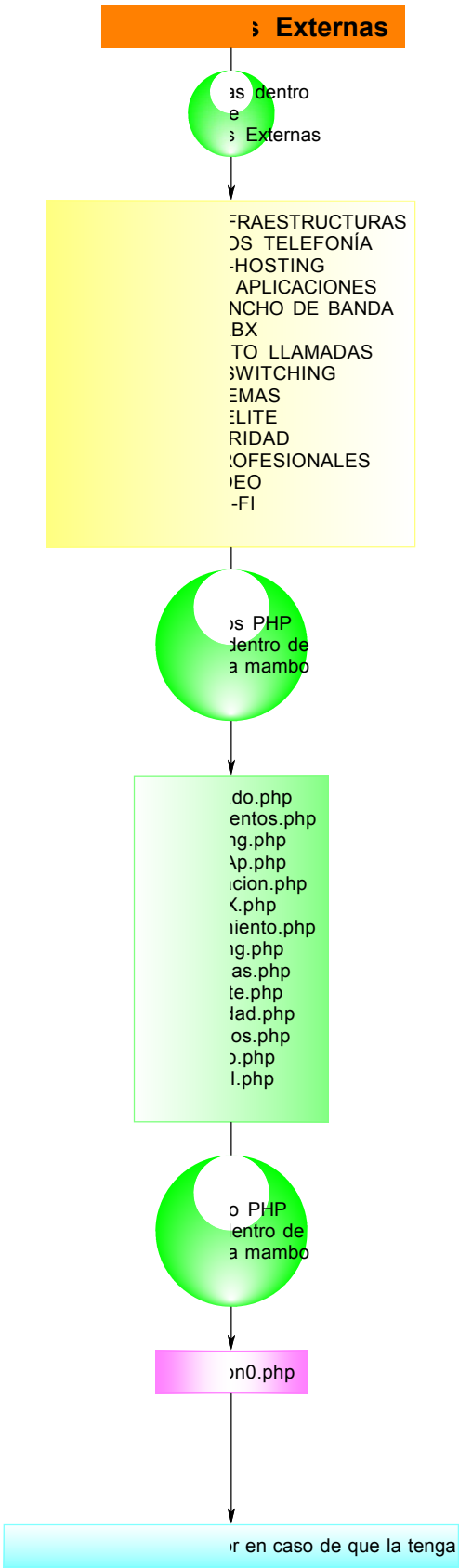


Figura 43: Clases PHP involucradas en la sección "Soluciones Externas".

El esquema representa los archivos PHP para el desarrollo de la sección “Soluciones Externas”. Dentro de esta sección hay 14 categorías y por cada una de ellas existe un fichero PHP correspondiente. El nombre de este archivo PHP tiene relación con la categoría para que sea lo más orientativo posible. Todos estos archivos deben de ser colocados dentro de la carpeta “./mambo”. Al seleccionar cualquiera de estas secciones aparece un contenido dónde se indica que se debe pinchar sobre una ficha para ver el listado de posibles soluciones relacionadas con esa tecnología.

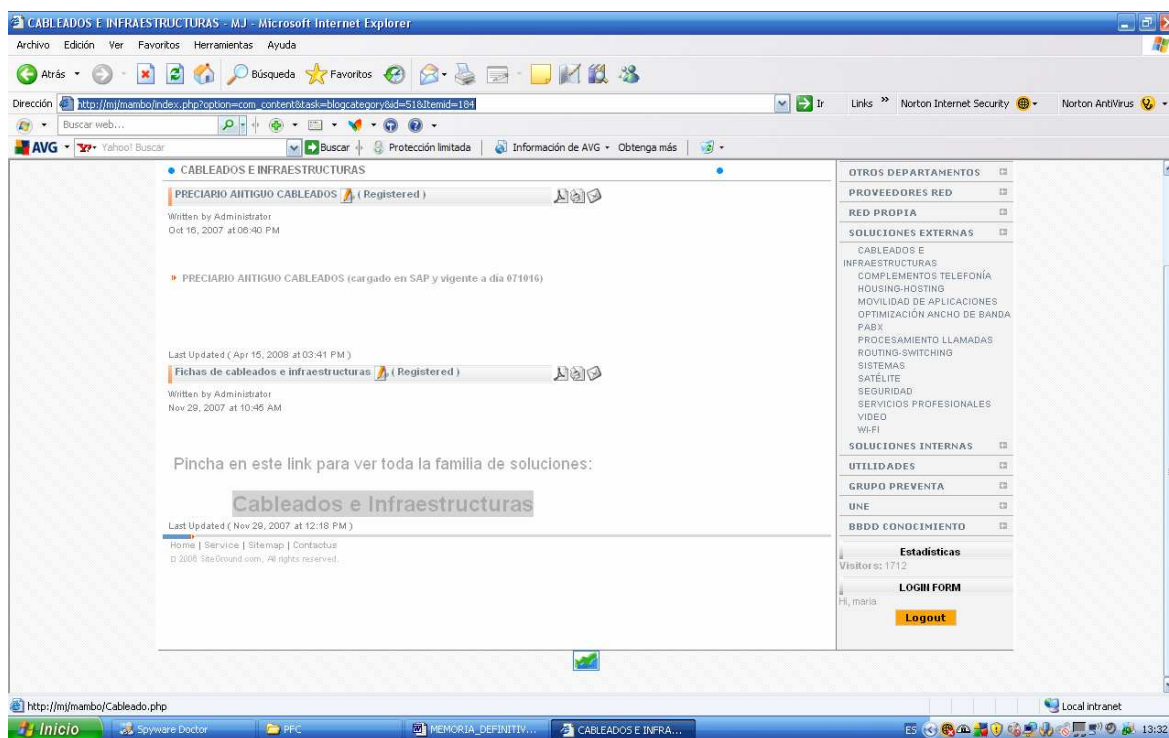


Figura 44: Vista de enlace que lleva al archivo PHP.

Una vez pulsado este botón se abre en pantalla completa un listado de todos los integradores con los que cuenta la empresa y que pueden dar servicio a esa solución. Por cada integrador hay un recuadro con los campos: Integrador, Escenario de aplicación, Productos relacionados, Ejemplo precios, Documento Técnico Tipo, Oferta Técnica Tipo, Configurador Precios y Tipo Compras. Para diferenciar un integrador de otro la cabecera está sombreada en color gris y el nombre del integrador es un enlace.

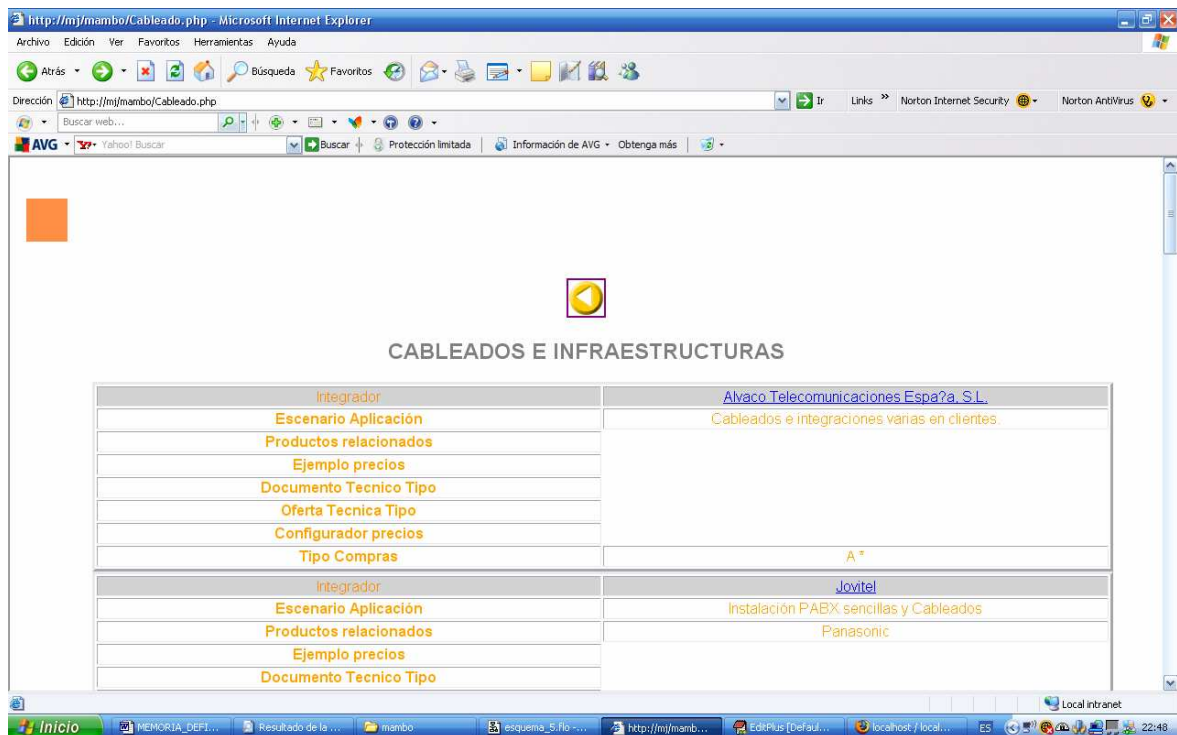


Figura 45: Listado de soluciones de la categoría "CABLEADOS E INFRAESTRUCTURAS"

Una vez que se tiene claro cuál es el integrador seleccionado para una oferta se puede pulsar el enlace que aparece en la cabecera. Este enlace lleva a otra página con toda la información específica de este integrador. Los datos que aquí aparecen son: enlace a la página Web de su empresa, datos para contactar con comerciales y técnicos, listado de otros productos de este integrador y un campo de comentarios.

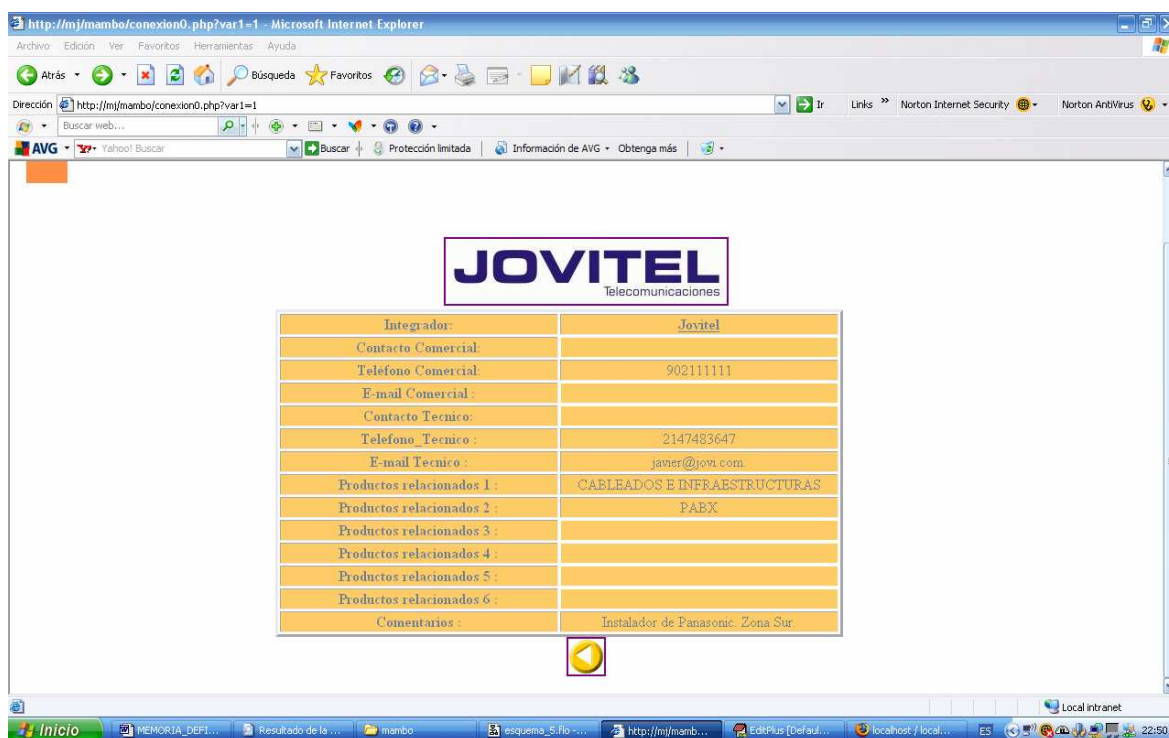


Figura 46: Datos del Integrador seleccionado.

Cuándo se ve esta pantalla se está ejecutando la clase de php **Integrador.php**. Este script recibe de la página que le invoca (**Cableado.php**, **Housing.php**...) una variable (var1) que indica el integrador elegido. Para enviar la variable de la página con el listado de integradores se utiliza la sentencia:

```
<a href="http://mj/mambo/conexion0.php?var1=<?php echo $valor1 ?>">
```

Es decir, esta sentencia debe aparecer en **Cableado.php**, **PABX.php**... Todas ellas invocan al mismo script PHP, **conexion0.php** para mostrar los valores de un integrador en concreto. Aquí el valor que identifica el integrado seleccionado se recoge con la sentencia:

```
$integrador=$_GET['var1'];
```

Realmente PHP no obliga a que el nombre de la variable dónde se envía y se recibe sea el mismo. Es decir seguiría funcionando igual si donde se envía se denominase var7, en lugar de var1. Tampoco es necesario que desde el script donde se envía la variable haya tenido que ser definida previamente al envío del valor. Es decir, funciona correctamente si

la primera vez que aparece la variable es en la sentencia del envío anteriormente comentada. Todos estos scripts tienen una hoja de estilo asociada. Los ficheros correspondientes a las categorías (**Cableado.php**, **Satelite.php**...) tienen asociada la misma hoja de estilo: **hoja_estilo.css**. En ella se define la cabecera, el tipo de botón para volver atrás que aparece 2 veces: al principio y al final de la lista de posibles integradores, el formato de las tabla de integradores...etc. El archivo **conexion0.php** tiene asociada la hoja de estilo: **hoja_estilo_int.css**.

BBDD Conocimiento

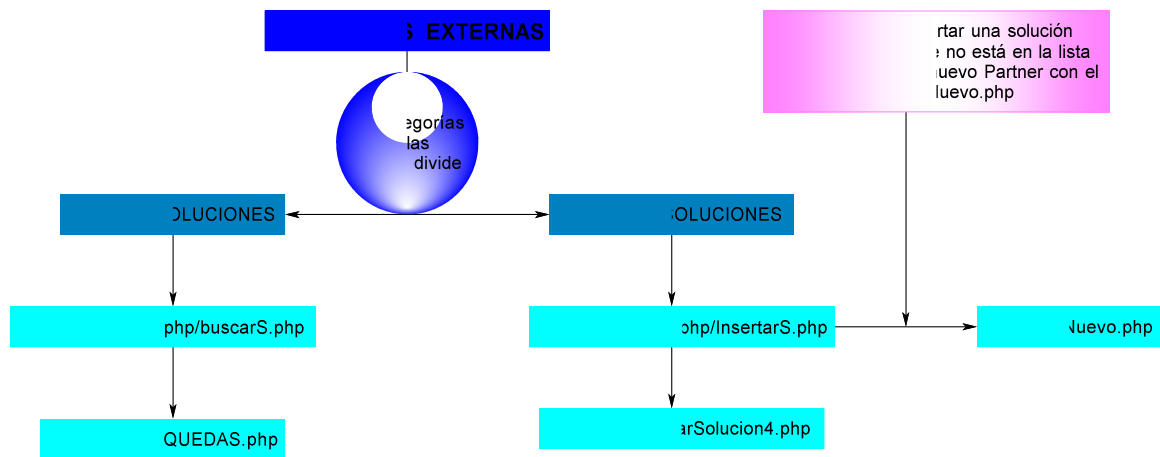


Figura 47: Ficheros PHP involucrados en la categoría "Soluciones Externas"

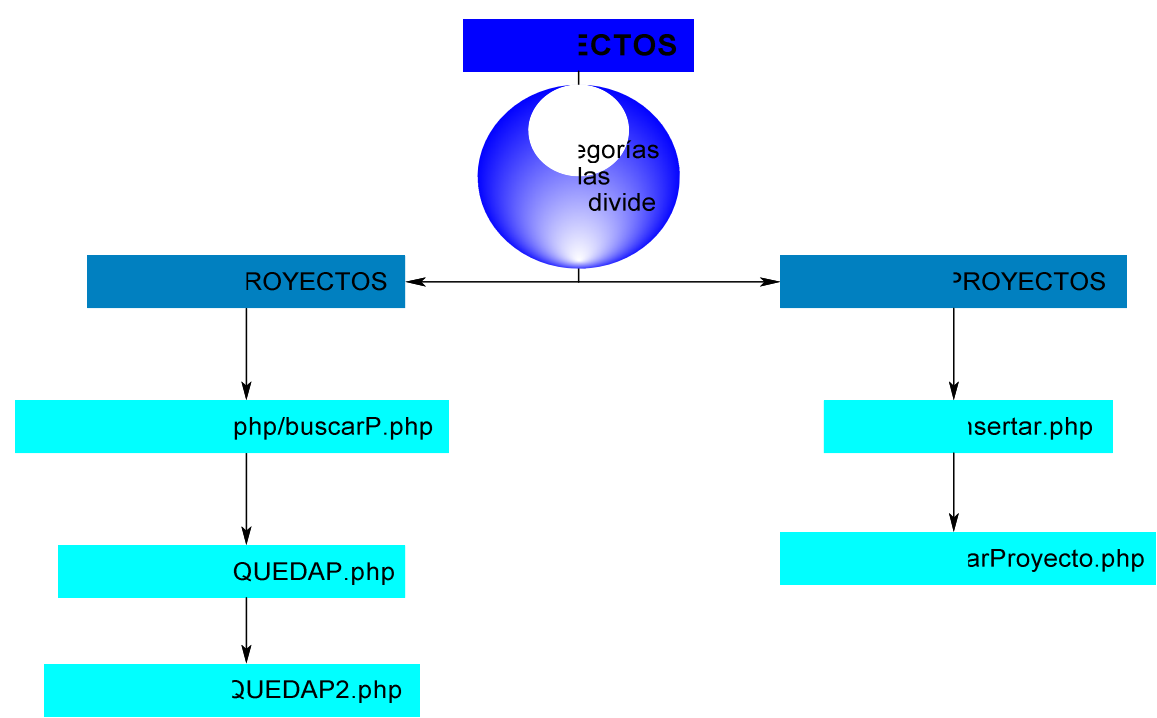


Figura 48: Ficheros PHP involucrados en la categoría "Proyectos".

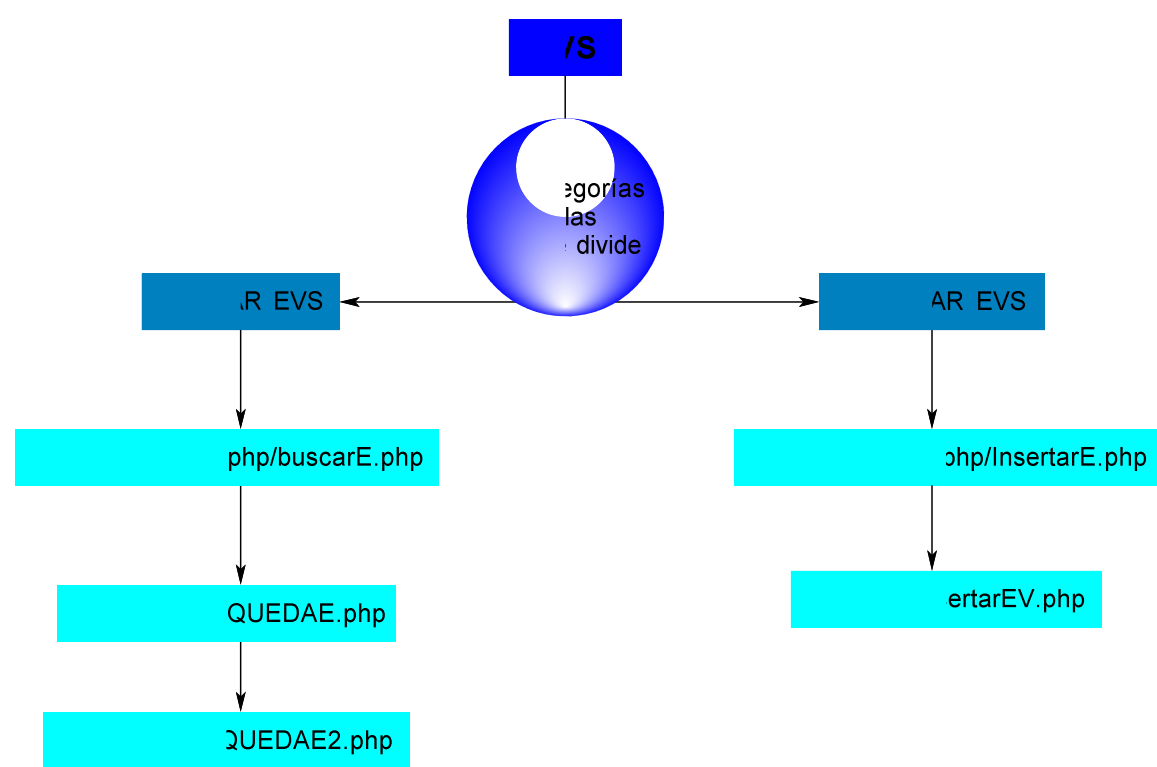


Figura 49: Ficheros PHP involucrados en la categoría "EVS".

Los esquemas anteriores representan los scripts PHP que entran en juego en la sección “BBDD CONOCIMIENTO”. Esta sección se divide en 3 categorías y cada una de ellas a su vez en 2 subcategorías: buscar e insertar. La estructura en las 3 es paralela. Es decir el código y la lógica es el mismo, la diferencia radica en la tabla de la base de datos dónde se busca o se inserta. La principal diferencia entre las 3, es que la subcategoría “INSERTAR PROYECTOS” no aparece incrustada directamente en el contenido, sino que tienes que pulsar un botón y aparece una pantalla completa para poder hacer la inserción. Esta excepción se hizo por una cuestión de formato, puesto que para insertar proyectos se deben introducir una mayor cantidad de campos. Se comprobó que si este formulario se incrustaba directamente en el código, la información no quedaba demasiado clara.

La opción de buscar es análoga a la sección “SOLUCIONES EXTERNAS” anteriormente comentada. La diferencia radica en que en la anterior sección, cuándo se hacía la búsqueda sobre la base de datos la palabra a buscar se escogía en el momento en que se pulsaba una u otra categoría. Es decir si se pulsaba la categoría VIDEO en la base de datos se buscaba por esta palabra. En “BBDD CONOCIMIENTO > SOLUCIONES EXTERNAS” si se busca VIDEO se obtienen los mismos resultados que si se hace a través de “SOLUCIONES EXTERNAS> VIDEO”. Puede parecer que al tener la misma funcionalidad sea redundante tener ambas opciones en la Intranet. Sin embargo la diferencia radica en que la parte de “SOLUCIONES EXTERNAS” se hace cargo de las soluciones más comunes, mientras que en la segunda parte, a parte de poder buscar esas soluciones también se pueden buscar otras soluciones que también están en la base de datos, pero que no son tan comunes. Así en el momento en que se inserte una nueva solución en la base de datos , se podrá buscar a través de BBDD independientemente que pertenezca o no a las soluciones más comunes.

Referencias

- [1] Apache Friends – XAMPP, <http://www.apachefriends.org/es/xampp.html>, [Visitado el 04/07/09]

- [2] Babin, Lee. Introducción a AJAX con PHP (PROGRAMACIÓN), ANAYA MULTIMEDIA, 2007, [Visitado el 23/01/09]

- [3] Características de Drupal, <http://drupal.org/es/caracteristicas>, [Visitado el 02/07/09]

- [4] Dev-Postnuke.com, <http://www.dev-postnuke.com/>, [Visitado el 05/07/09]

- [5] Douglas Paterson, Learning Mambo. A Step-by-Step Tutorial Building Your Website., 2006, [Visitado el 04/07/09]

- [6] [EasyPHP]- PHP-Apache- MySQL- PhpMyAdmin for Windows, <http://www.easyphp.org/index.php>, [Visitado el 04/07/09]

- [7] Firtman, Maximiliano. AJAX: WEB 2.0 PARA PROFESIONALES, MARCOMBO S.A., 2008, [Visitado el 29/01/09]

- [8] Gilfillan, Ian., LA BIBLIA DE MySQL, ANAYA MULTIMEDIA 2003 [Visitado el 02/05/09]

- [9] GLOSARIO DE INTERNET, <http://www.uco.es/ccg/glosario/glosario.html#INTERNET> , [Visitado el 04/07/09]
- [10] Install PHP 5 Apache MySQL on Windows: WampServer, <http://www.wampserver.com/en/> , [Visitado el 04/07/09]
- [11] Intranets Corporativas, <http://dominiopublico.com/intranets/> [Visitado el 04/07/09]
- [12] Introducción a Microsoft Office SharePoint Server 2007, <http://office.microsoft.com/es-es/sharepointserver/HA101732173082.aspx> , [Visitado el 02/07/09]
- [13] Joomla! <http://forum.joomla.org/> [7] Intranets Corporativas, <http://dominiopublico.com/intranets/> [Visitado el 04/07/09]
- [14] La Web del Programador, <http://www.lawebdelprogramador.com/> [Visitado el 07/08/09]
- [15] Mambo Hispano, <http://www.mambohispano.org/> [Visitado el 05/07/09]
- [16] OpenCms, the Open Source Content Management System/CMS, <http://www.opencms.org/en//> , [Visitado el 05/08/09]
- [17] Orange Business Services, <http://www.orange-business.com/es/index.html> [Visitado el 06/07/09]

- [18] Planeta CMS, <http://www.planetacms.com/> , [Visitado el 06/07/09]
- [19] PHP: Hypertext Preprocessor, <http://www.php.net/> [Visitado el 25/03/09]
- [20] PHP a fondo, <http://www.desarrolloweb.com/php/> [Visitado el 26/08/09]
- [21] PHP-Hispano.NET, <http://www.php-hispano.net/> [Visitado el 02/08/09]
- [22] Programación Web- Programación y diseño de Páginas Web, <http://www.programacionweb.net/> [Visitado el 14/08/09]
- [23] Python Programming Language, <http://www.python.org/> , [Visitado el 14/08/09]
- [24] Raya, José Luis y Raya, Laura. INTRANETS Y TCP-IP CON MICROSOFT WINDOWS SERVER 2003, RA-MA, 2004, [Visitado el 17/07/09]
- [25] RedIRIS-Cortafuegos: Conceptos teóricos, <http://www.rediris.es/cert/doc/unixsec/node23.html>, [Visitado el 17/07/09]
- [26] SourceForge.net: Find and Develop Open Source Software, <http://sourceforge.net/>, [Visitado el 04/07/09]
- [27] The Apache Software Foundation, <http://www.apache.org/> ,[Visitado el 04/07/09]

- [28] The Perl Directory, <http://www.perl.org/> ,[Visitado el 04/07/09]
- [29] The Uniform Server, <http://www.vivaphp.com.ar/soft/uniform-server-3.3.html> ,[Visitado el 04/07/09]
- [30] Tobias Hauser. Christian Wenz, Mastering Mambo. E-Commerce, Templates, Module Development, SEO, Security and Performance. 2005 [Visitado el 04/07/09]
- [31] Zope.org, <http://www.zope.org/> , [Visitado el 05/07/09]